APPLIED PROBABILISTIC INFERENCE: MODEL ESTIMATION FOR HVAC PREDICTIVE CONTROLS AND ALL-WEATHER PERCEPTION FOR AUTONOMOUS VEHICLES

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Peter Paul Radecki

May 2016

APPLIED PROBABILISTIC INFERENCE: MODEL ESTIMATION FOR HVAC PREDICTIVE CONTROLS AND ALL-WEATHER PERCEPTION FOR AUTONOMOUS VEHICLES

Peter Paul Radecki, Ph.D.

Cornell University 2016

Probabilistic inference and reasoning is applied to two major application areas: HVAC controls in buildings and autonomous vehicle perception. Although the physical domains differ vastly, across both applications the presented novel contributions share real-time inference of stochastic systems for improved control capability and performance. Besides performing simple state estimation, Kalman Filters in both applications are extended for model inference—estimating thermal model parameters and disturbances in buildings and dynamic object classification for perception in autonomous vehicles.

Part one of this study proposes a general, scalable method to learn control-oriented thermal models of buildings that could enable wide-scale deployment of cost-effective predictive controls. An Unscented Kalman Filter augmented for parameter and disturbance estimation is shown to accurately learn and predict a building's thermal response. By leveraging building topology and measurement data, the filter quickly learns parameters of a thermal network during periods of known or constrained loads and then characterizes unknown loads in order to provide accurate 24+ hour energy predictions. Performance was validated with EnergyPlus simulation data across a year-long study of a passive building.

The method is extended to multi-zone actively controlled buildings by using

the controller to excite unknown portions of the building's dynamics. A simulation study demonstrates self-excitation improves model estimation. Formalization of parameterization, disturbance estimation, and self-excitation routines is shown with an observability analysis. Comparing against a baseline thermostat controller, a Model Predictive Control (MPC) framework, which anticipates weather uncertainty and time-varying temperature set-points, is shown to improve energy savings and occupant comfort.

Part two of this study presents a novel probabilistic perception algorithm as a real-time joint solution to data association, object tracking, and object classification for an autonomous ground vehicle (AGV) in all-weather conditions. The presented algorithm extends a Rao-Blackwellized Particle Filter originally built for Cornell's AGV for the DARPA Urban Challenge (DUC) to include multiple model tracking for classification. Additionally a state-of-the-art vision detection algorithm that includes heading information for AGV applications was implemented. Cornell's AGV from the DUC was upgraded and used to experimentally examine if and how state-of-the-art vision algorithms can complement or replace lidar and radar sensors. Sensor and algorithm performance in adverse weather and lighting conditions is tested. Experimental evaluation demonstrates that sensor diversity with a joint probabilistic perception algorithm provides robust all-weather data association, tracking, and classification.

BIOGRAPHICAL SKETCH

Peter Radecki was born in El Paso, Texas but grew up in Michigan's Upper Peninsula. He has held summer internships at Megavolt Inc. of Springfield Remanufacturing Corporation, in the Structural Dynamics Summer School at Los Alamos National Laboratory, and an NSF REU in fluid power at Georgia Institute of Technology. While a student at Michigan Technological University, Peter enjoyed playing on MTU's club hockey team, snowboarding at Mont Ripley, and being the Guidance, Navigation, and Controls team leader for the Oculus-ASR – MTU's winning entry to the 6th University Nanosat Program. In 2009 Peter graduated Magna Cum Laude from Michigan Tech with his B.S. degree in mechanical engineering with minors in mathematics, computer science, and electrical engineering. Peter worked at MIT Lincoln Laboratory as a co-op before pursuing graduate school. In 2010 Peter received Honorable Mention from the NSF Graduate Research Fellowship Program, Finalist Alternate from the DOE Science Graduate Fellowship, and was awarded three years of funding from the National Defense Science and Engineering Graduate (NDSEG) Fellowship. In 2013 Peter received his M.S. degree in mechanical engineering from Cornell University. His research interests include controls and estimation applied to real-world systems. In his rare moments of spare time he enjoys hockey, Catholic literature, woodworking, and wrenching on cars. Peter currently holds an engineering development position with General Motors Powertrain at the Milford Proving Grounds in Michigan.

To my parents, Pete and Sheila Radecki

ACKNOWLEDGEMENTS

I would like to acknowledge all those who instructed, challenged, and motivated me along the journey to pursue an engineering Ph.D.

Most importantly I would like to sincerely thank both of my advisors Prof. Mark Campbell and Dr. Brandon Hencey for their patience, direction, and guidance through the years it took to conduct my research and write my papers and dissertation. Thanks also to Professor Noah Snavely for being part of my dissertation committee and generously giving his time and expertise. I would also like to acknowledge Professor Mark Psiaki, Professor Hadas Kress-Gazit, and the late Professor Ephrahim Garcia for their insightful courses and advice on research, outreach, and life after grad school.

Interacting with all of the fellow students at Cornell made the time in Ithaca very enjoyable. First off I would like to acknowledge Kevin Matzen for collaboration on the Skynet perception project and Justin Dobbs for collaboration on several HVAC research projects. I would especially like to thank my girlfriend, Monica VanderPutten, for proofreading this entire dissertation and motivating me to finish it. Additionally I would like to acknowledge Kevin Wyffels, Alex Ivanov, Lucas de la Garza, Mike Meller, Rodrigo Zeledon, Matt Leineweber, Karen Chiang, Ryan Mitch, Monica Nguyen, Radhika Patel, Kevin Kircher, Bryan Peele, Jose Rios, and Rick Pampuro for all of their camaraderie, academic advice, and encouragement during my time as a student at Cornell. Lastly I would like to acknowledge all the folks who have worked on Skynet over the years – that platform enabled my last research paper.

I would like to especially thank Professor Gordon Parker for letting me work in his Intelligent Control Systems Lab at MTU while I was still in high school – that early exposure to controls and research was the biggest reason why I pur-

sued mechanical engineering and eventually doctoral research. Also at MTU, I thank Professor Brad King for his leadership of the Aerospace Enterprise and nanosatellite program and challenging me to pursue a Ph.D., and I thank Professor Chuck Van Karsen and Professor Jason Blough for their guidance on various sensor and vibration projects while I was an undergraduate student.

I would also like to thank several of my previous internship and co-op bosses. I would like to thank Professor Mykel Kochenderfer for showing me the CS side of controls and estimation, Professor Wayne Book for teaching me how to balance the theoretical and applied sides of a research project, and Dr. Gyuhae Park for teaching me how to write an academic paper.

Lastly I would like to dedicate this work to my parents. Their hard work and Faith raised me with the curiosity and discipline to pursue an engineering doctorate. Any successes I have achieved are due in large part to how they raised me and the advice I continue to receive from them.

This work was supported by the Department of Defense (DoD) through the National Defense Science & Engineering Graduate Fellowship (NDSEG) Program.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Probabilistic inference and reasoning is applied to two major application areas: HVAC controls in buildings and autonomous vehicle perception. Although the physical domains differ vastly, across both applications the presented novel contributions share real-time inference of stochastic systems for improved control capability and performance. Besides performing simple state estimation, Kalman Filters in both applications are extended for model inference—estimating thermal model parameters and disturbances in buildings and dynamic object classification for perception in autonomous vehicles.

Recent studies of buildings' heating, ventilating, and air-conditioning systems have shown significant energy savings are possible with advanced model predictive control systems. A scalable cost-effective method to readily acquire accurate, robust models of individual building's unique thermal envelope has historically been elusive and hindered the widespread deployment of prediction-based control systems. Continuous commissioning and lifetime performance of these thermal models requires deployment of on-line data-driven system identification and parameter estimation routines.

The work in Chapter 2, submitted for review to the IEEE Transactions on Control Systems Technology journal, proposes a general, scalable method to learn control-oriented thermal models of buildings that could enable wide-scale deployment of cost-effective predictive controls. An Unscented Kalman Filter augmented for parameter and disturbance estimation is shown to accurately learn and predict a building's thermal response. We propose a novel gray-box approach using an Unscented Kalman Filter based on a multi-zone thermal net-

work and validate it with EnergyPlus simulation data. The filter quickly learns parameters of a thermal network during periods of known or constrained loads and then characterizes unknown loads in order to provide accurate 24+ hour energy predictions. Initial results of this study were published in the 2012 American Controls Conference under the title "Online Building Thermal Parameter Estimation via Unscented Kalman Filtering" [1]. This study extends the initial investigation by formalizing parameter and disturbance estimation routines and demonstrating results across a year-long study.

Chapter 3 investigates a method to improve buildings' thermal predictive control performance via online identification and excitation (active learning process) that minimally disrupts normal operations. An initial, study published in the 2013 IEEE Conference on Decision and Control under the title "Online Thermal Estimation, Control, and Self-Excitation of Buildings" [2], was extended with formalization of the active learning process and has been submitted to the Energy and Buildings journal. Using multi-zone actively controlled buildings we examine how to improve the thermal model estimation by using the controller to excite unknown portions of the building's dynamics. Comparing against a baseline thermostat controller, a Model Predictive Control (MPC) framework, which anticipates weather uncertainty and time-varying temperature set-points, is shown to improve energy savings and occupant comfort. By coupling building topology, estimation, and control routines into a single online framework, we have demonstrated the potential for low-cost scalable methods to actively learn and control buildings to both ensure occupant comfort and minimize energy usage, all while using the existing building's HVAC sensors and hardware.

The biggest challenge remaining for autonomous ground vehicles (AGV) is robust perception. The architecture for a prototypical autonomous ground vehicle is not yet set in terms of sensing, computation hardware, or programming interfaces. The major perception components commonly found in AGV systems are generally known as segmentation and clustering (processing raw sensor analog or digital data into obstacle-level meta-measurements), data association (determining what measurement came from what static or dynamic environmental obstacle), tracking (estimating the obstacles state, position, velocity, etc.), and classification (distinguishing cars, people, buildings, etc.). Despite a decade of advancement since the original DARPA Grand Challenge, weather performance remains nebulous. To date, joint probabilistic solutions to the entire perception problem have been stated as intractable to formulate or compute and much research has focused on advancement of individual perception components in isolation from the overall solution. One of the keys to handling adverse environmental operating conditions will be a full Bayesian probabilistic joint perception system thereby minimizing the number of brittle ad hoc design choices which tend to fail under uncommon untested weather scenarios.

In Chapter 4, a novel probabilistic perception algorithm is presented as a real-time joint solution to data association, object tracking, and object classification for an autonomous ground vehicle in all-weather conditions. The work is currently in preparation for submission to the Journal of Field Robotics. The presented algorithm is based on a Rao-Blackwellized Particle Filter originally built for Skynet, Cornell's AGV for the DARPA Urban Challenge. The original algorithm, containing a particle filter for data association and a Kalman filter for multi-object tracking, is extended to additionally include multiple model tracking for classification. Additionally a state-of-the-art vision detection al-

gorithm that includes heading information for AGV applications was implemented. Cornell's AGV from the DARPA Urban Challenge was upgraded and used to experimentally examine if and how state-of-the-art vision algorithms can complement or replace lidar and radar sensors. Sensor and algorithm performance in adverse weather and lighting conditions was tested. Experimental evaluation demonstrates robust all-weather data association, tracking, and classification where camera, lidar, and radar sensors complement each other inside the joint probabilistic perception algorithm.

CHAPTER 2

ONLINE MODEL ESTIMATION FOR PREDICTIVE THERMAL CONTROL OF BUILDINGS

2.1 Introduction

Significant energy savings in buildings' heating, ventilating, and air-conditioning (HVAC) systems could be realized with advanced control systems [3], but deployment of these control systems requires a method to readily acquire low cost models of buildings' unique thermal envelopes [4], [5]. Previous studies have investigated several methods but generally arrived at non-scalable specialized solutions [6], [7], [8], [9], [10].

Ideally, a Building Automation System (BAS) would automatically modify set-points and load shedding based on weather, occupancy, and utility pricing predictions [11]. Every building has unique and time-varying thermal dynamics, occupancy, and heat loads which must be characterized accurately if a BAS is to apply model predictive controllers (MPC) to realize energy and monetary savings [5]. Additional considerations include: measured building data often contains low information content; engineering models contain designer's intent instead of actual construction; and building's usage evolves over time [12]. Unfortunately, in practice there has yet to be demonstrated a scalable, low-cost method to readily acquire these much needed accurate models of individual buildings' unique thermal envelopes.

For continuous commissioning and lifetime adaptability a low-cost scalable method to acquire control-oriented building models must: learn both the dy-

namics and the disturbance patterns quickly, provide stable extrapolation, be adaptable to future changes in building structure or use, and use existing available data. White-box, first-principles, forward modeling approaches are often inaccurate, not robust to changes, and take extensive engineering or research effort to build [4], [13]. Recent advances, such as those by Braun, have shown methods to generate control-oriented models from accurate high fidelity white-box models, but difficulties remain in acquiring these high fidelity models [14], [15]. Black-box approaches take up to 6-months to train and cannot be safely extrapolated into new control configurations [16], [17]. In [1] we demonstrated the first published study of a scalable modeling and online estimation framework for multi-zone building states, parameters, and unmodeled dynamics using a gray-box estimation technique. Since our study in 2012 several other researchers have validated our initial claims and highlighted new challenges for learning control-oriented building models with gray-box methods [7], [8], [9], [11], [15], [18], [19], [20], [21], [2].

Continuing our work in [1], we propose a multi-mode Unscented Kalman Filter (UKF) as a generalizable on-line gray-box data-driven method to learn the building's multi-zone thermal dynamics and detect unknown time varying thermal loads. By coupling known building information and simple physics models with existing measurable building data we demonstrate how a probabilistic estimation framework can adapt over time to continually learn both dynamics and disturbances while providing stable prediction performance.

Building upon recent advances by other researchers previously highlighted, this paper aims to generalize our findings and method with the following contributions:

- development of minimal parameterization for dynamics estimation,
- generalized thermal disturbance pattern estimation,
- multi-mode heuristic for simultaneous parameter and disturbance estimation.

This paper represents the basis for a scalable method to acquire control-oriented building models that are adaptable for future use and provide stable extrapolation for new control configurations.

The paper is organized as follows. After formulating an extensible on-line data-driven parameter estimator using an UKF, the main contributions of the paper—minimal parameterization, generalized thermal disturbance estimation, and multi-mode learning heuristic—are shown in the Thermal Estimation Approach section. Based on data generated from our simple 5-zone thermal model plus a more complex 5-zone EnergyPlus simulated model, less than 2 weeks of training data is shown to make reliable 24 hour predictions. Utility and robustness of the UKF is then demonstrated across a year-long study. Based on testing and performance, a discussion of how the UKF fits into the building thermal modeling problem and an identification of areas of future research conclude the paper.

2.2 Thermal Estimation Formulation

This section formulates the parameter estimation problem for building energy models. A sufficiently accurate model is crucial for successful implementation of model-based control methods, and robust online adaptation is necessary to

accommodate life-cycle and seasonal changes of the building and its environment.

2.2.1 Building Energy Model

In general, we consider an explicit building energy model taking the form

$$\dot{x} = f(x, u_1, u_2, p) \tag{2.1}$$

$$z = h(x, u_1) \tag{2.2}$$

where the states x relate energy and mass quantities (e.g. temperature and humidity), u_1 are measured/known inputs (e.g., heater output), u_2 are unmeasured inputs (e.g., solar load, plug load, infiltration), p are selected uncertain model parameters, and z is the measured output (e.g. zone temperature/humidity).

Naturally, this model is designed to approximate the dominant phenomena of a true building energy system. For this paper, a thermal network model is used to represent heat and mass transfer among nodes of a graph [8], [11], [18], [22]. For the sake of simplicity, this paper makes well-accepted simplifying assumptions for radiative heat transfer and mass transfer—a simple linear approximation enables it to be lumped with conductive heat transfer [23]. Similarly, this study assumes a low-humidity, heating dominant climate, and thus neglects humidity. However, the framework could readily be augmented to include the effects of nonlinear radiative heat transfer and humidity [24].

Convection, conduction, and mass transfer heat flux (watts) into zone i is contributed from the temperature differential relative to connected adjacent zone(s) j divided by the thermal resistance R_{ij} (degree/watt) plus an additive

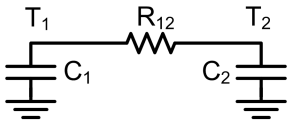

Figure 2.1: Two node example thermal network.

disturbance term b_i (watts). *(Note: unless otherwise mentioned, subscripts denote zones.)* The heat flux and thermal capacity C_i (joule/degree) affects the time-based temperature rate of change $\dot{T}_i$. Thus the temperature rate of change of zone i due to connection(s) with zone(s) j and disturbance b_i is

$$\dot{T}_i = \sum_j (T_j - T_i)/(R_{ij}C_i) + b_i/C_i \tag{2.3}$$

The derived representation for temperature change due to heat transfer is mathematically analogous to voltage change due to current flow in a resistor-capacitor network. For visualization, a simple 2-node example with two capacitances and one resistance is shown in Fig. 2.1.

Using the 2-node example, a state space representation can be derived where $\bar{T}$ is a vector of temperatures; A is a matrix of RC values; and $\bar{b}(t)$ is a vector of additive, independent, time-varying disturbances such as solar radiation.

$$\begin{aligned} \dot{\bar{T}}_i &= A\bar{T}(t) + \bar{b}(t) \\ A &= \begin{bmatrix} -\frac{1}{R_{12}C_1} & \frac{1}{R_{12}C_1} \\ \frac{1}{R_{12}C_1} & -\frac{1}{R_{12}C_1} \end{bmatrix} \\ \bar{b}(t) &= \begin{bmatrix} \frac{b_1(t)}{C_1} \\ \frac{b_2(t)}{C_2} \end{bmatrix} \end{aligned} \tag{2.4}$$

Based on [25], an n-node thermal network can be formalized by defining a simple undirected weighted graph with: nodes $N := \{1, 2, \cdots, n\}$ that are assigned capacitances C_i and temperatures T_i; edges $E \subset N \times N$ that connect adjacent nodes; and weights $\{R_{ij} \forall (i, j) \in E : R_{ij} = R_{ji}\}$ that are assigned resistances. For a general thermal network with n nodes, the A matrix is

$$A = \begin{cases} A_{ij} = 0, & \text{if} i \neq j, (i, j) \notin E \\ A_{ij} = \frac{1}{C_i R_{ij}}, & \text{if} i \neq j, \in E \\ A_{ij} = -\sum_{l \neq i} A_{il}, & \text{if} i = j, \end{cases} \tag{2.5}$$

2.2.2 Estimation Problem

A short explanation of our proposed method follows. Using a simple first order heat transfer model with multiple zones, the UKF estimates model parameters of the thermal dynamics during periods which have small or well-characterized thermal loads. After learning the dynamics during low disturbance periods, such as nighttime, the UKF is augmented to track unknown disturbances while continuing to improve its dynamics model. The UKF internally maintains a covariance quality metric which only adjusts parameter estimates if the incoming data provides new thermal information.

Thermal disturbances significantly affect most buildings but are often overly complex to model, requiring information about building geometry and neighboring foliage [26]. Solar gain is treated as an unmodeled external disturbance. This simplification removes complexities of modeling diffuse and direct sunlight, shading, and night sky radiation temperature, and allows for simple dis-

turbance generation in EnergyPlus by turning on or off environmental radiation transfer. The solar gain provides us with a specific periodic disturbance to estimate with patterns. In practice this technique could estimate any number of disturbances if one has some information about the disturbance frequency, intensity, or timing such as dusk and dawn times or building occupancy times. Common examples amenable to disturbance pattern estimation include occupant body-heat, equipment, computers, electrical loads, lighting, and HVAC.

For all tests the state space system is integrated at one minute interval time-steps with Euler integration to allow discrete-time filter implementation. Parameter estimation with the Kalman Filter is achieved by augmenting the temperature states $\bar{T} = \begin{bmatrix} T_1 & \cdots & T_n \end{bmatrix}^\top$ with unique parameters $\bar{p} = \begin{bmatrix} (RC)_1 & \cdots & (RC)_k \end{bmatrix}^\top$ and disturbances $\bar{b} = \begin{bmatrix} (b/C)_1 & \cdots & (b/C)_l \end{bmatrix}^\top$ together in the state representation $\hat{x} = \begin{bmatrix} \bar{T}^\top & \bar{p}^\top & \bar{b}^\top \end{bmatrix}^\top$. For the purposes of estimation, the full discrete-time stochastic system is

$$\begin{aligned} \bar{T}(k+1) &= A(p(k))\bar{T}(k) + \bar{b}(k) + \bar{w}_1(k) \\ \bar{p}(k+1) &= \bar{p}(k) + \bar{w}_2(k) \\ \bar{b}(k+1) &= \bar{b}(k) + \bar{w}_3(k) \\ \bar{z}(k) &= \bar{T}(k) + \bar{v}(k) \end{aligned} \tag{2.6}$$

where $\bar{w}_1(k)$ represents process noise, $\bar{w}_2(k)$ represents estimation uncertainty in RC parameters, $\bar{w}_3(k)$ represents process noise for disturbances, and $\bar{v}(k)$ represents measurement noise. Artificial process noise for the constant parameters, denoted $\bar{w}_2(k)$, allows the filter to change its estimate of these values through time and allows the filter to track the true time varying disturbance. The set of process noise terms $\bar{w}_1(k)$, $\bar{w}_2(k)$, and $\bar{w}_3(k)$ are stacked as defined by the state $\hat{x}$.

All noise terms are assumed zero mean, Gaussian, white, and stationary. This representation results in multiplication and division of estimated parameters through the dynamics function. Specifically, temperature is being multiplied by RC parameters necessitating non-linear estimation techniques.

Measurement noise is specified based on the accuracy of the temperature sensors. Process noise is specified for the temperature states based on the level of zone aggregation used while the RC and disturbance process noise is set to an artificial value greater than zero in order to allow the filter to vary its estimate of these parameters through time. Increasing process noise level for any parameter indicates that the model isn't confident of its ability to describe the process evolution of that parameter. Disturbances, which by their nature the model is not explicitly capturing, are biased and vary with time. In order to estimate the disturbances over time, their noise level is set to be non-zero. Because the RC values should be fairly constant, while the disturbance bias may change throughout the course of a day, the noise level for RC parameters should be much smaller than for disturbances.

2.2.3 Parameterization Problem

A minimal set of independent parameters must be specified for filters to enforce the system dynamics during parameter estimation [27]. Over-parameterization causes unidentifiable parameter manifolds or extra degrees of freedom and can result in violation of dynamics constraints and physics laws such as conservation of energy. In machine learning and system identification, indeterminate degrees of freedom can cause overfitting where the model learns the noise in-

stead of the dynamics of interest. In estimation theory, parameter observability requires that the Fisher information matrix be invertible—redundant parameters or over parameterization breaks this observability criterion resulting in an unobservable subspace [28].

Efficient and reliable parameter estimation requires estimating a minimal number of parameters [27]. From Equation (2.4) there are only two unique parameters required to describe the A matrix despite it containing three variables—two resistances and one capacitance. The extra parameter acts as a scaling factor and can be quantified only if the heat flux q is provided in addition to the temperature histories. Without the scaling factor only a time-constant can be inferred. This ambiguity generally makes the estimation problem numerically unstable, theoretically unobservable, or practically unreliable. Rectifying the ambiguity could be done with actual heat flux information which is generally unavailable in practice, so for this study, selecting a minimal set of parameters mitigates the problem.

2.3 Thermal Estimation Approach

2.3.1 Parameterization

For any thermal network without heat flux information, the total number of unique parameters is one less than the sum of the total number of resistances and capacitances. RC products are estimated together in order to reduce the non-linearity of the estimation problem. However, by estimating RC products, extra parmaters are mathematically introduced into the estimation problem due

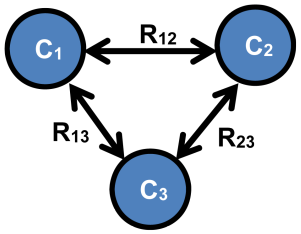

Figure 2.2: Three-node graph with one loop.

to the multiplicative combinations. A careful graph study generated methods to obtain a minimal parameter set for thermal network estimation.

Two nodes that have no shared conduction or convection are considered independent, and any edge directly connecting them is pruned from the graph to give the simplest representation. Independent ambient nodes such as external temperatures have infinite capacitance in the thermal network. External nodes may have unique update functions depending on the simulation and weather desired for the modeling exercise.

Because diagonal terms in A are linear combinations of the off-diagonal terms, parameter estimation is only performed for off-diagonals. Parameterization of trees, graphs with no cycles of which Fig. 2.1 is an example, with combined RC products automatically guarantees a minimal representation of the system.

Unfortunately this minimal guarantee does not extend to graphs containing closed cycles. Fig. 2.2 is an example of a graph containing a cycle whose state space A matrix is shown.

$$A = \begin{bmatrix} -(\frac{1}{R_{12}C_1} + \frac{1}{R_{13}C_1}) & \frac{1}{R_{12}C_1} & \frac{1}{R_{13}C_1} \\ \frac{1}{R_{12}C_2} & -(\frac{1}{R_{12}C_2} + \frac{1}{R_{23}C_2}) & \frac{1}{R_{23}C_2} \\ \frac{1}{R_{13}C_3} & \frac{1}{R_{23}C_3} & -(\frac{1}{R_{13}C_3} + \frac{1}{R_{23}C_3}) \end{bmatrix} \tag{2.7}$$

We arbitrarily select $R_{13}C_1$ to show that one of the six RC products is redundant and can be eliminated by multiplying and dividing the other $R_{ij}C_i$ parameters by each other around the cycle:

$$R_{13}C_1 = \frac{R_{12}C_1 \times R_{23}C_2 \times R_{13}C_3}{R_{12}C_2 \times R_{23}C_3} \tag{2.8}$$

In a graph, each cycle which uses at least one unique edge and passes through no nodes with infinite capacitance may be used to eliminate one redundant RC product from the estimation problem by multiplying and dividing around the loop. In general unique edges should be selected for elimination. Eliminating a shared edge between two cycles joins the two cycles mathematically through multiplication in the estimation routine which can negatively impact numerical stability. Selecting multiple redundant parameters to prune from a graph estimation problem should be done such that each redundant RC parameter lies on a globally unique edge for its respective cycle, and the shortest available cycle should be chosen for calculation in order to guarantee minimal parameter cross-sensitivity.

The process is best understood through illustration. The building in Fig. 2.4 contains 5 internal zones with 8 resistances plus one external zone (not drawn) with 5 resistances connecting to the internal zones. The external zone is assumed to have infinite capacitance and controlled by a weather forcing function.

This model contains 13 resistances and 5 capacitances, less one, gives 17 unique parameters to estimate from 21 RC products. As an example, unique edge $\{1, 2\}$ can be used in cycle $\{1, 2, 5\}$ to multiplicatively cancel out an RC product, namely $R_{12}C_1$ or $R_{12}C_2$. The 4 edges $\{1, 2\}$, $\{2, 3\}$, $\{3, 4\}$, $\{4, 1\}$ are all unique and can each be used to eliminate one RC product in order to reduce the number of RC products to 17 to match the number of unique parameters for the estimation problem.

2.3.2 Disturbance Estimation and Pattern Recognition

Direct sensing of disturbance heat flux is rarely practical in building systems. However, timing information for disturbances is typically available, so a practical method is presented to learn disturbances given only timing information with no prior disturbance quantification. This method could readily be augmented if additional disturbance heat flux data was available.

Looking at the thermal model in Equation (3.3), a change in zone temperature may be explained away using either connected zones with their respective temperatures or additive disturbances. In order to get satisfactory estimation performance this ambiguity must be considered in the filter design lest it manifest problems akin to non-minimal parameterization. The study by Fux [18] also discussed the concept of multi-mode learning and the importance of characterizing disturbances. Our engineering solution splits the estimation problem based on the presence of disturbances and manipulates the process covariance for estimated disturbance parameters based on timing of expected unquantified disturbances. From a control theory perspective the system does not have time-invariant observability. However, buildings are time-varying systems that have

some periodicity. Looking over a horizon, for example one day for solar disturbances, we can learn constant parameters when no disturbances are present and learn disturbances after having estimated the constant parameters. This partitioning enables time-varying observability.

The presented method is not claimed to be optimal, rather it is a practical solution based on engineering judgment of typical scenarios common in buildings. Typically, a system can sense if people are using a building but cannot measure their heat flux or the equipment they use, likewise it can sense if the HVAC system is on but not the exact heat flow delivered to a specific room. The approach attempts to use commonly available timing knowledge to quantify and infer disturbances that are not directly measurable and only partially predictable, such as u_1 or u_2 from Equation (3.1).

Learning of disturbances is done in a Markov fashion: the estimator assumes no knowledge of previous historical disturbance patterns and estimates a new disturbance value b_i at each time step based on the previous time step's estimate, the dynamic model, and the current measurement. The disturbance states in the UKF are modeled as constants which have zero-mean Gaussian additive noise. A characteristic change in the disturbance, such as a heater turning on or the sun coming up at dawn, violates the zero-mean assumption causing a bias in the disturbance. In order to track these sudden bias changes using a simple UKF, the variance(s) correlating to those specific zone(s) disturbances are inflated to allow the filter to acquire and track the new value. This artificial tuning of the covariance is similar to tuning a forgetting factor in adaptive control frameworks.

In the Matlab-based simulation, heating or cooling is arbitrarily added to in-

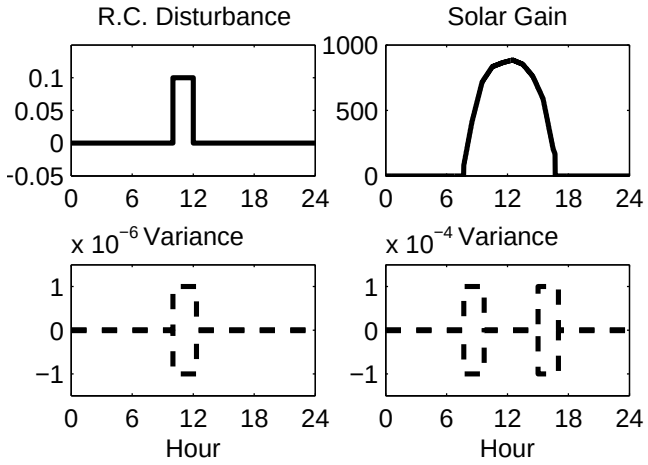

Figure 2.3: Disturbance Parameter Process Variance Tuning.

dividual zones from 10am to noon, so the covariance is increased around those times. In the EnergyPlus simulation, the primary unmodeled disturbance is solar radiation, so the covariance is increased around dusk and dawn. This variance tuning is visually depicted in Fig. 2.3.

Because the UKF can explain temperature swings by either tuning RC or disturbance values, a multi-mode approach provides a uniform method to split the estimation problem. In order to acquire good estimates, the UKF is operated in two modes: A) Acquisition Mode: initially only RC products are estimated by running the filter at night when solar gains are at a minimum, and B) Monitoring Mode: both RC products and disturbances are estimated simultaneously after RC product estimates have started to converge to constant values.

Disturbance estimates from the entire multi-day learning period are heuristically combined in order to generate a 24-hour pattern. This disturbance pattern is then used when predicting the building's thermal response. The heuristic pattern recognition algorithm is a simple weighted average described in Algorithm 1.

Algorithm 1: Disturbance pattern estimation and prediction

1: **procedure** PATTERNESTIMATION
2: Given:
3: *bias(t,i)* ← estimated bias per iteration
4: *weight(i)* ← disturbance scaling factor
5: *timing(i)* ← disturbance timing
6: *bias(k,i)* ← **TimeNormalize**(*bias(t,i)*, *timing(i)*)
7: *dist(k)* ← **WeightedAverage**(*bias(k,i)*, *weight(i)*)
8: **procedure** PREDICTION
9: Given:
10: *weight(i)* ← predicted scaling factor
11: *timing(i)* ← predicted timing
12: *bias(k,i)* ← **Scale**(*dist(k)*, *weight(i)*)
13: *bias(t,i)* ← **TimeUnNormalize**(*bias(k,i)*, *timing(i)*)

The predominant unmodeled disturbance from our EnergyPlus data is radiation from the sun, which is dependent on cloud cover, time of day, season of year, and other factors. For the purpose of engineering a robust simple solution we make a realistic assumption that we have a measurement of the average solar intensity for morning and afternoon and use this for both pattern recognition and simulation predictions. The solar intensity reading is calculated as the summation of the Environment Direct Solar and Environment Diffuse Solar variables from EnergyPlus.

By averaging the solar intensity before midday and after midday, two weights are determined for each day. In order to ensure sufficient signal to

noise ratio for disturbance pattern estimation, a day's disturbance estimates are discarded if the day's total solar intensity averaged below 35% of the maximum solar intensity possible for that location.

2.4 Simulation Results

2.4.1 UKF 5-room Simulated Performance

A six-node thermal network, corresponding to five internal zones and one external temperature shown in Fig. 2.4, is used to evaluate the UKF parameter estimation and thermal disturbance detection. For this first evaluation, measurement data is generated from a model whose dynamics are structurally identical to the dynamics used in the UKF. The 5-room models shown here are based on that in [1], but feature extended explanations, derivations, and simulation results.

Given five finite capacitances and thirteen resistances, there are a total of seventeen unique RC products for the UKF to estimate in this model. The external temperature forcing function with controlled frequency content is composed from the sum of a 40 degree peak-to-peak sinusoid with period of one day, a 10 degree peak-to-peak sinusoid with period of 4 hours, and random noise which allows the temperature to drift from day to day. Resistance and capacitance values were chosen such that thermal lag in the simulation would be similar order as thermal lag in a small to medium sized building.

Daily repeating disturbances, which turned on at 10:00 and off at 12:00, are

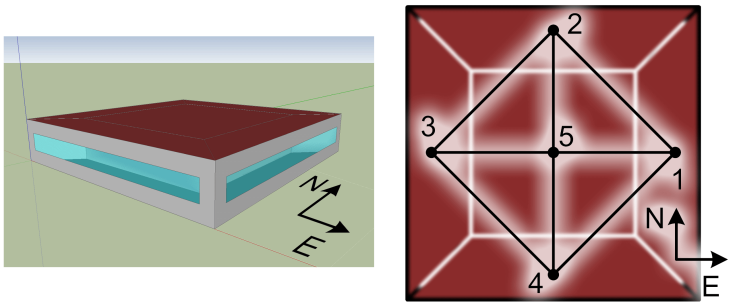

Figure 2.4: Left: Five-room building used for simulations, Right: Top view of node labeled internal thermal network representation. White lines represent building internal walls. A sixth unlabeled node acts as an external temperature.

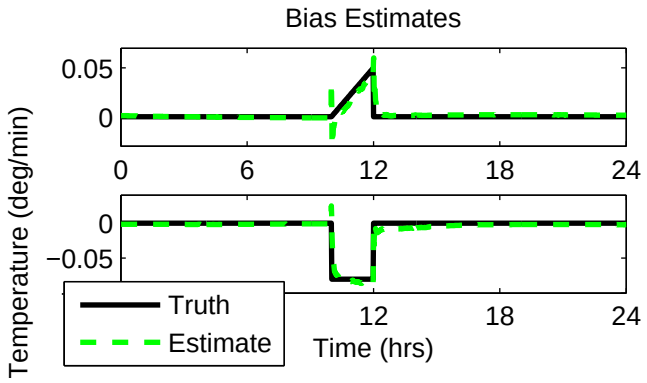

Figure 2.5: The actual disturbances introduced into system for the two zones shown in Fig. 2.6 are labeled above as "Truth" and were repeated on a 24 hour cycle. Dashed lines represent the UKF 4-day average estimate of the disturbances which was used for predictions in Fig. 2.6. (Deg F)

introduced uniquely into each zone of the house. For the combined RC and disturbance estimation, 4 days of data is used for training. With the variance tuning method and pattern recognition, the filter has excellent disturbance tracking; example tracking of two disturbances are shown in Fig. 2.5. A 48-hour prediction is then made using the average disturbance and final RC estimates along with the exact external temperature profile. In a real system inaccuracies

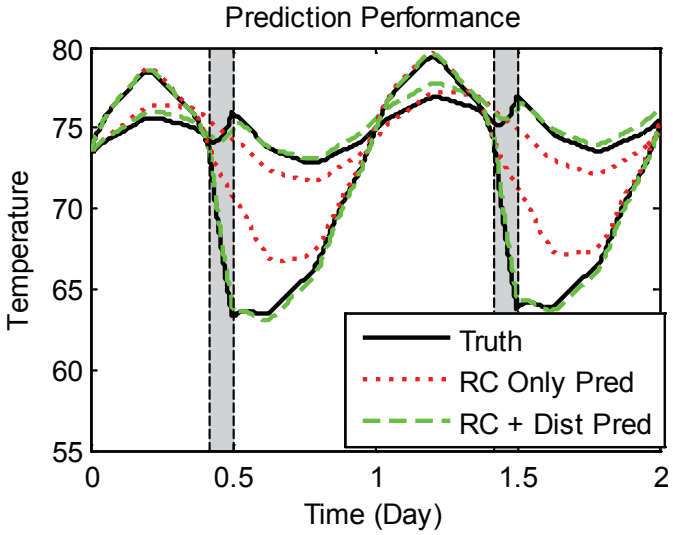

Figure 2.6: Example 48-hour prediction derived from RC estimates and estimated daily cyclic disturbances. Gray boxes denote times where external disturbances were present in the true system. An external temperature forcing function was composed from two sinusoids plus low amplitude white noise. (Deg F)

in the weather forecast will degrade the prediction quality. Excellent predictions were made with the estimated model. Fig. 2.6 plots temperatures of two zones, comparing predictions from the RC only estimation model and the RC plus disturbance estimation model to the truth model. This evaluation provides a good indication of the applicability of the UKF to thermal network parameter and disturbance estimation.

2.4.2 UKF EnergyPlus Performance

Given the excellent performance of the UKF on data generated by the thermal network model, the UKF was tested on data generated from an EnergyPlus simulation. Individual data traces show typical estimation performance which is validated in an aggregated year-long demonstration. This more realistic Ener-

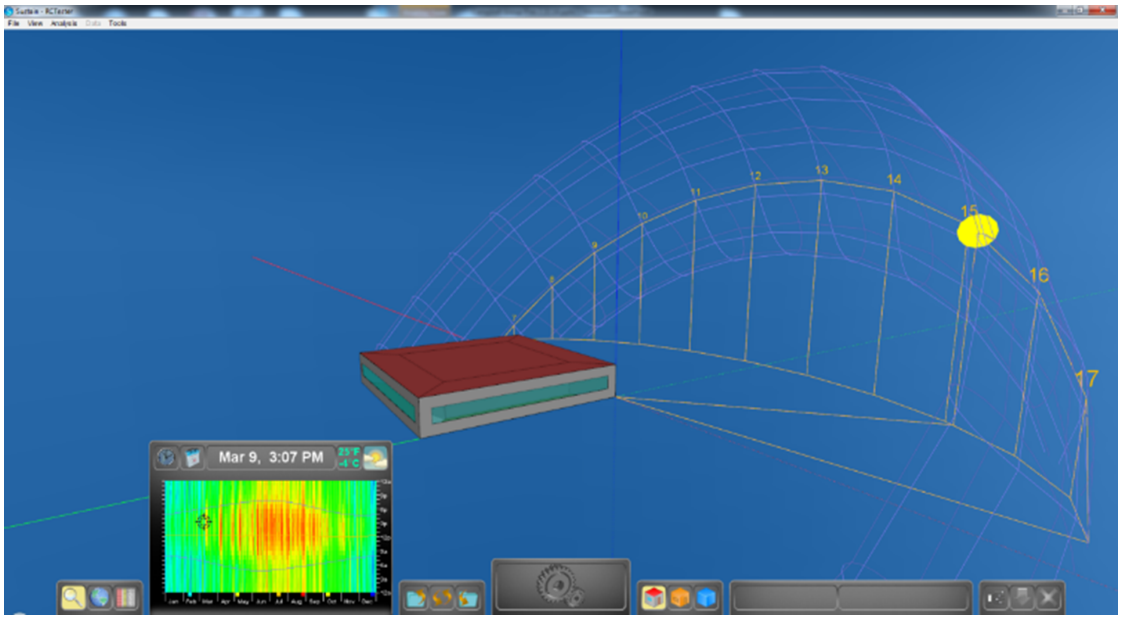

Figure 2.7: Screenshot from Sustain showing afternoon sun on West Wall.

gyPlus model, shown in Fig. 2.4, has five rooms correlating to the five zones; realistic data for the floor, wall, and ceiling composition; and windows on the four exterior walls, which point in the cardinal compass directions. The structure is simulated with weather and solar radiation for Elmira, NY.

Fig. 2.7 shows the path of the sun on March 9th and the variability of the sun path over the course of the year that is simulated by EnergyPlus. The graphic was generated by Sustain, a front-end for EnergyPlus developed by researchers at Cornell University Program of Computer Graphics [29]. Fig. 2.9 shows the resulting average disturbances from a four day test where biases are only estimated for the last two days and then combined into a 24-hour pattern. Notice how the East room is heated in the morning, the West room is heated in the afternoon, and the South room is heated all day, which correlates nicely with expected heat from solar radiation. Variations present in disturbance estimates,

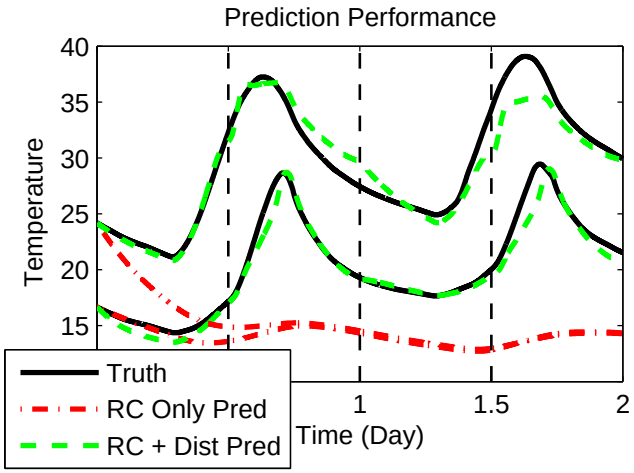

Figure 2.8: (Top) South Zone, (Bottom) West Zone of 48-hour EnergyPlus predictions (Deg C).

such as the peak during the 12th hour in the South room, are due to actual weather variations present in the short two day learning period. Estimating disturbances over longer periods of time smooths out these variations. Predictions utilize the final estimated RC parameters and 24-hr disturbance patterns, zone initial temperature conditions, predicted external temperature profile, and half-day average predicted solar intensity. Plots of the 48-hour predictions are shown in Fig. 2.8. Predictions which utilize solar disturbances have much higher accuracy than the RC only predictions. The accuracy of these predictions ground assumptions made in the thermal network formulation and more importantly demonstrate the utility of the UKF for system identification of buildings' thermal envelope.

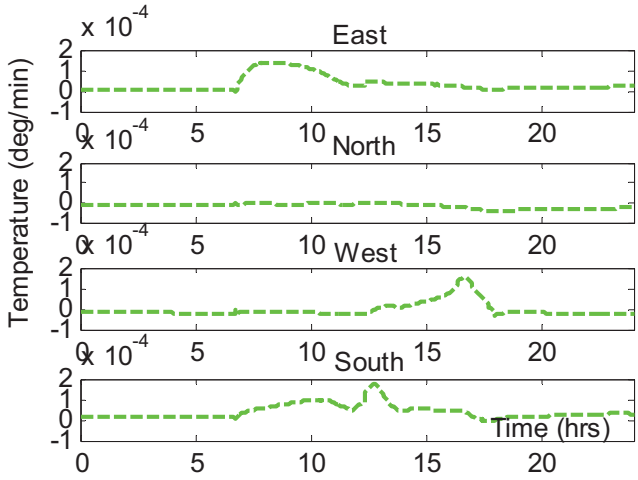

Figure 2.9: Estimated 24-hour cyclic disturbances from EnergyPlus data (Deg C).

2.4.3 UKF EnergyPlus Year Study

Further analysis of the EnergyPlus generated data was conducted by analyzing a total year of data. A unique UKF instance is initialized each day on the first 357 days of the year and run for 7 days, 3 days in acquisition mode and 4 days in monitoring mode. Then the UKF predicts the building's response on the eighth day and is compared against the building's actual response from EnergyPlus. This generated 357 sets of learned parameters, bias estimates, and 24-hour prediction simulations.

Of the total set, 43 simulations resulted in estimation routine errors such as negative parameter estimates or covariance shrinking to zero causing an UKF matrix inverse calculation failure. From further analysis, simple estimation monitoring by a human or addition of heuristic rules to the existing framework would fix all 43 estimation routine failures. For example over 10 of the failures occurred because 4 consecutive days had less than 35% solar radiation causing no disturbance pattern to be learned. Fixing these sorts of numerical issues to

guarantee 100% reliability in an automated algorithm is outside the scope of the current study. Results suggest the algorithm could easily be matured for practical application by adding a number of heuristics.

Using the 314 successful estimation runs, we compared the 24-hour prediction simulations against the building's truth simulation from EnergyPlus and found good accuracy—the models have enough fidelity to be used for control. Over a 12 hour prediction horizon the root mean square (RMS) temperature prediction error was 1.16°C and over 24 hours the RMS temperature prediction error was 1.48°C. ASHRAE standards mandate that vertical temperature stratification in an occupied zone should be less than 5.4°F (3°C) [30]. Home and office thermostats often use a dead-band of 4°F (2.2°C) to 8°F (4.4°C). The model's prediction errors are well within these design bounds for the 24 hour prediction horizon. In Fig. 2.10 the RMS error for the prediction is shown over time demonstrating good performance. Increased learning periods of 14 and 28 days further reduced prediction error but did not drive it to zero—likely because the simple RC model could not capture the entire fidelity of the EnergyPlus truth simulation.

Additionally, Fig. 2.11 shows a month-long stable prediction of the East and West zones based on a model learned from 7 days of data. This long-horizon prediction uses the same data types as that in Fig. 2.8: correct zone initial temperature conditions, correct external temperatures over the horizon and half day average solar intensity values over the horizon. The long-horizon prediction accuracy demonstrates the learned model is unbiased, stable, and robust. To the author's knowledge, this is the first year-long study of an online UKF estimating disturbances with parameters and states for a building.

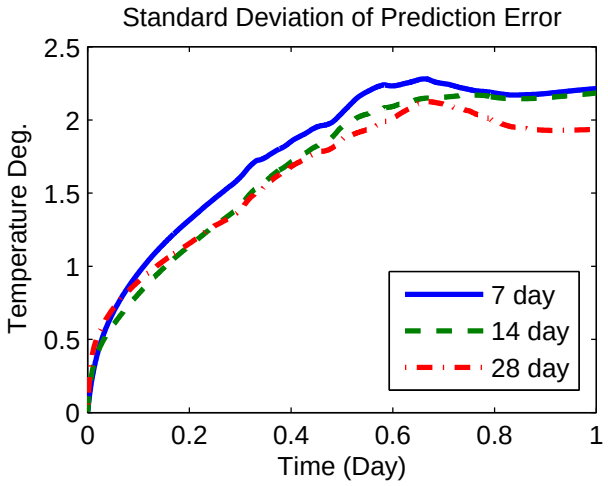

Figure 2.10: Root Mean Square error of temperature predictions of all 5 zones over the year for different lengths of estimation learning time (Deg C).

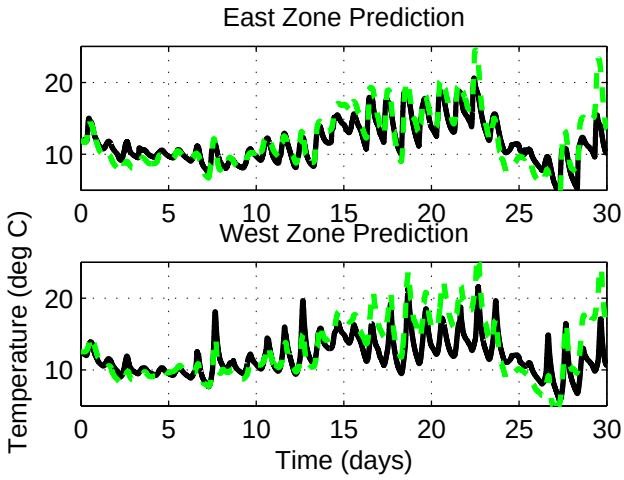

Figure 2.11: Month prediction using a model learned from 1 week of data.

2.5 Discussion

Results of the UKF estimation and model prediction capabilities have demonstrated the method as a powerful tool for thermal modeling of building systems. The simplicity with which a thermal network can be described combined with the numerical stability and robustness of the UKF are important factors which

could enable its deployment as a scalable system identification routine for buildings' thermal envelopes.

No physics constraints were applied to ensure RC parameters were positive, or to inform the bias and disturbance estimation. Realistic estimation of values was solely dependent on the quality of the chosen thermal model representation, estimation technique, and measured data. The authors expect that good results obtained from this paper's simulations would reflect realistic expectations of good performance in real world applications given gray-box estimation demonstrations using real data in other recent studies of specific applications [18], [19], [22]. In this paper accurate bias tracking was achieved though covariance tuning, but this might not be scalable to certain buildings where disturbances occur on erratic schedules, so multiple hypothesis estimation or constraints may be augmented with a UKF to provide a more powerful solution [20], [21].

Further investigation into model selection and fidelity could lead to performance improvements for the UKF depending on the target application and available computation and sensing hardware. For example, Dobbs [31] compared accuracy of thermal models across different levels of RC zone aggregation—leveraging such a tool may aid control-oriented model creation for the UKF. Extensions to the UKF may offer new opportunities for fault detection and monitoring [7].

One outstanding challenge remaining with this online estimation technique is a demonstration of the learned models' performance with model predictive controllers in practice. Some studies [19], [2], [9] have begun investigating how the quality of the learned model affects the performance of predictive controllers

that use the model. The consensus to date is that intra-zone excitation is necessary in order to learn a building's internal coupling. Before controlling the building in a novel way to maximize energy savings, a building's internal thermal coupling must be known. Deriving methods to monitor the quality of the measured data and better learn the building's thermal dynamics on-demand, by experimentally exciting the building, are the subject of a future paper by the authors.

2.6 Conclusion

A multi-mode implementation of a multi-zone UKF was presented as a scalable and rapidly deployable system identification routine for building thermal dynamics. Using a 5-room model, the UKF demonstrated the ability to learn both dynamics parameters for a thermal network and unknown disturbances. 24-Hour predictions from UKF estimated parameters yielded accurate results which were validated with EnergyPlus simulations using a full year of data. The UKF, a data-driven, model-based approach, amenable to augmentation with numerical methods, provides a promising step towards a scalable framework to realize advanced BAS predictive controllers.

Acknowledgment

We would like to thank Dave Bosworth for his help generating an EnergyPlus 5-room building dataset for evaluation of parameter estimation methods.

CHAPTER 3

SELF-EXCITATION: AN ENABLER FOR ONLINE THERMAL ESTIMATION AND MODEL PREDICTIVE CONTROL OF BUILDINGS

3.1 Introduction

Improved control methods have extraordinary potential for energy savings within building climate systems [3]. To date, the diversity of building climate systems configuration, usage, and heating/cooling equipment challenges the scalability and cost effectiveness of improved control methods [4]. Specifically, improved control generally necessitates a more detailed understanding and insight into the system's behaviors. Unfortunately, the data typically collected from buildings, which lack adequate excitation, often does not contain enough information to learn a control-oriented predictive model [32], [33], [5], [34], [9]. Based on our previous work [2], by adding self-excitation via HVAC actuators, we propose a managed online framework for model estimation and active learning that is scalable to a wide range of buildings.

It is widely recognized that obtaining the process model is the most time consuming task in application of advanced model-based control techniques. This paper focuses on methods to acquire a building model. Passive learning, identification, or adaptation requires sufficient excitation [35, §5.2]. In-situ measurement data often does not contain enough information, so active learning and identification must excite the control signals to increase the information content in measured data. However, blind excitation can be disruptive and must be intelligently executed. In summary, we desire methods that meet following criteria (a) pinpoint crucial or potentially detrimental model uncertainties; (b) excite

only modes that reduce crucial or potentially detrimental model uncertainties; and (c) maintain excitation within acceptable, non-disruptive operational limits.

As has been investigated by numerous researchers [36], [37], [38], [38], [39], [40], [41] Model Predictive Control (MPC) provides a practical method to minimize energy while ensuring occupant comfort in the presence of changing weather and disturbance uncertainties. Unfortunately, due to the difficulty in obtaining an accurate model for both the thermal dynamics and disturbances of buildings, model predictive controllers are typically absent from Building Automation Systems (BAS) [12], [4]. The level of insight, complexity, and tuning required to deliver a high performance BAS is labor and expertise intensive; therefore, heuristic and intuition-driven approaches are not scalable or economical for most buildings [42]. Furthermore, poor disturbance, thermal, or occupancy models can cause MPC to perform worse than a typical thermostat as temperature predictions are sensitive to model parameter accuracy [37], [43], [44], which necessitates acquiring accurate thermal models and disturbance models of buildings.

Passive learning and identification techniques take many forms but typically do not have data with sufficient information content for accurate model inference. Typical learning methods fall into various categories such as offline or batch model fitting [45], Recursive Least Squares [46, 33, 1], stochastic grey-box models [47, 19], and adaptive control techniques [18]. If the data provided to these methods does not contain information content necessary to infer an accurate model, any passive technique is relegated as not applicable.

Learning the dynamics of any system from measured data requires excitation of the given system; without sufficient excitation, the system parameters

are unobservable. Privara highlighted three main challenges in data-driven modeling for MPC: 1) data violates typical persistent excitation requirements, 2) increased model complexity increases length of learning time and suitable experiments may be expensive, 3) measured temperature signals are often co-linear [32]. As detailed throughout the adaptive control literature, without sufficient excitation, catastrophic parameter estimate drift may occur [35, §5.2 & §8.3]. Excitation must be sufficient to achieve the signal to noise ratio and information content required for parameter estimation. Active excitation techniques can reduce plant/model mismatch by using input signals to actively excite and perturb the system. System identification techniques design excitation signals using sum of sinusoids or pseudo-random binary inputs [48], [49], [50], [51]. Similarly, extremum seeking uses a dithering signal to estimate the objective function gradient [52]. In general these brute force excitation techniques can be costly and disruptive to the desired system operation. As a result, optimal experiment design seeks to minimize model uncertainty and excitation power or energy [53].

Various excitation schemes have incorporated prior knowledge of the system and examined ways to improve the information content of measured data. Building systems are part of a class of poorly excited systems [54]; a Bayesian framework provides a natural way to incorporate prior information [34]. Active learning methods use prior information to focus excitation on uncertain aspects of the model. However, such techniques lack a feedback loop that adapts to incorporate information accumulated from system operation [55] and minimize the cost and disruption of excitation.

Techniques are needed to balance the cost of exploration with the benefits of

exploitation, which is a common theme in reinforcement learning [56]. The classic example is multi-armed bandit problem, the essence of which asks: should the current estimate be exploited for currently best-known control, or should the current estimate be improved via exploration such that the improved estimate can be exploited for future improved control. The overall problem can also be viewed from a Design of Experiments (DOE) perspective where we wish to design optimal or efficient experiments that extract the most new information possible while consuming the least energy, minimally disrupting occupants, and not violating any physical hardware constraints [57]. In an operational building, induced excitation is often necessary—due to poor zone segregation, bad signal to noise ratio, or poor data observability—but difficult to manage without wasting energy or annoying occupants [32]. Li, who wrote one of the most recent extensive literature surveys on building modeling for control and operation, concluded that generation of appropriate excitation signals while considering the characteristics and constraints of building energy systems is an urgent research topic [5]. Studies to date have demonstrated the importance of self-excitation and shown the benefits of a learned model coupled to MPC. However, a comprehensive framework that learns over time and automatically identifies appropriate self-excitation has yet to be presented.

We propose a framework for online self-excitation of the HVAC system that provides the information necessary to learn and update building models while not wasting energy, unnecessarily disrupting occupants, or harming equipment. By building upon our initial study presented in [2] the main contributions of this paper include:

- Development of an Experiment Generator that determines, based on the

current model, which zones should be excited;

- Development of an Experiment Selector that automatically selects and runs experiments while obeying energy and occupant constraints;
- Observability analysis confirming and explaining the basis for our excitation approach.

The Experiment Generator and Selector are wrapped into an overall on-line framework that includes an UKF and MPC for estimation and control.

We propose a framework to demonstrate the utility of coupling gray-box estimation methods with predictive controllers to create an online deployable BAS. After a brief explanation of the chosen underlying thermal model and Kalman Filter parameter estimate representation, we highlight the Model Predictive Controller with soft-constraint temperature bounds, excitation generation, and estimation monitoring. A 2-zone building simulation is developed to demonstrate the entire framework actively learning, improving occupant comfort, and reducing energy use compared to a baseline thermostat controller. An observability analysis sheds light on what mathematically occurs during self-excitation of the building. We conclude by discussing limitations of our framework and directions for future research.

3.2 Thermal Model Estimation

The thermal model, estimator, and controller shown below expand upon that presented in our early investigation [2] and provide a basis upon which to derive the online framework for self-excitation.

3.2.1 Building Energy Model and Parameterization

In general, we consider an explicit building energy model taking the form

$$\dot{x} = f(x, u_1, u_2, p, q) \tag{3.1}$$

$$z = h(x, u_1) \tag{3.2}$$

where the states x relate energy and mass quantities (e.g. temperature and humidity), u_1 are measured/known inputs (e.g., heater output), u_2 are unmeasured inputs (e.g., solar load, plug load, infiltration), p and q are selected uncertain model parameters, and z is the measured output (e.g. zone temperature/humidity).

Naturally, this model is designed to approximate the dominant phenomena of a true building energy system. For this paper, a thermal network model is used to represent heat and mass transfer among nodes of a graph [8], [11], [18], [22], [25]. For the sake of simplicity, this paper makes well-accepted simplifying assumptions for radiative heat transfer and mass transfer—a simple linear approximation enables it to be lumped with conductive heat transfer [23]. Similarly, this study assumes a low-humidity, heating dominant climate, and thus neglects humidity. However, the framework could readily be augmented to include the effects of nonlinear radiative heat transfer and humidity [24].

The temperature rate of change of zone i due to connection(s) with zone(s) j and disturbance b_i is

$$\dot{T}_i = \sum_j (T_j - T_i)/(R_{ij}C_i) + b_i/C_i \tag{3.3}$$

Convection, conduction, and mass transfer heat flux (watts) into zone i is contributed from the temperature differential relative to connected adjacent zone(s)

j divided by the thermal resistance R_{ij} (degree/watt) plus an additive disturbance term b_i (watts)—for this study it is assumed that each zone has independent heating control. *(Note: unless otherwise mentioned, subscripts denote zones.)* The heat flux and thermal capacity C_i (joule/degree) affects the time-based temperature rate of change $\dot{T}_i$. Following units through the multiplication of resistances (degree/watt) and capacitances (joule/degree) shows that RC products have units of seconds. The RC products represent time constants relating adjacent nodes in the RC network.

The derived representation for temperature change due to heat transfer is mathematically analogous to voltage change due to current flow in a resistor-capacitor network. This formulation can be expanded to build a full linear state space representation of the thermal dynamics

$$\begin{aligned} \dot{\bar{T}}(t) &= A\bar{T}(t) + B\bar{u}(t) \\ A &= \begin{cases} A_{ij} = 0, & \text{if} i \neq j, (i,j) \notin E \\ A_{ij} = \frac{1}{C_i R_{ij}}, & \text{if} i \neq j, \in E \\ A_{ij} = -\sum_{l \neq i} A_{il}, & \text{if} i = j, \end{cases} \\ B &= \text{diag}_{1\ldots n}\{b_i/C_i\} \end{aligned} \tag{3.4}$$

where each thermal zone is a node in the RC network. Matrix A represents the heat exchange between zones, $\bar{T}(t)$ is a vector of node temperatures, B represents the total heat output of the heater per zone, and $\bar{u}(t)$ is the control vector describing the fraction of the heater's output varying between 0 and 1. The formulation of matrix A is included because its structure is used later in the self-excitation explanation. For a general thermal network with n nodes, the A matrix can be constructed in Equation (3.4) as a simple undirected weighted graph with: nodes $N := \{1, 2, \cdots, n\}$ that are assigned capacitances C_i and temperatures T_i; edges

$E \subset N \times N$ that connect adjacent nodes; and weights $\{R_{ij} \; \forall \; (i,j) \in E : R_{ij} = R_{ji}\}$ that are assigned resistances. Lastly, the diagonal matrix B is defined per node by the maximum heater output b_i divided by the thermal capacitance C_i as shown in Equation (3.4). External temperature is modeled as a node in the thermal network with infinite capacitance whose temperature is modified by a weather forcing function.

For all tests the state space system is integrated at one minute interval time-steps with Euler integration to allow discrete-time filter implementation. Parameter estimation with the Kalman Filter is achieved by augmenting the temperature states $\bar{T} = \begin{bmatrix} T_1 & \cdots & T_n \end{bmatrix}^\top$ with unique parameters $\bar{p} = \begin{bmatrix} (RC)_1 & \cdots & (RC)_m \end{bmatrix}^\top$ and disturbances $\bar{q} = \begin{bmatrix} (b/C)_1 & \cdots & (b/C)_l \end{bmatrix}^\top$ together in the state representation $\hat{x} = \begin{bmatrix} \bar{T}^\top & \bar{p}^\top & \bar{q}^\top \end{bmatrix}^\top$. For the purposes of estimation, the full discrete-time stochastic system is

$$\begin{aligned} \bar{T}(k+1) &= A(\bar{p}(k))\bar{T}(k) + B(\bar{q}(k))\bar{u}(k) + \bar{w}_1(k) \\ \bar{p}(k+1) &= \bar{p}(k) + \bar{w}_2(k) \\ \bar{q}(k+1) &= \bar{q}(k) + \bar{w}_3(k) \\ \bar{z}(k) &= \bar{T}(k) + \bar{v}(k) \end{aligned} \tag{3.5}$$

where $\bar{w}_1(k)$ represents process noise, $\bar{w}_2(k)$ represents estimation uncertainty in RC parameters, $\bar{w}_3(k)$ represents process noise for disturbances, and $\bar{v}(k)$ represents measurement noise. Artificial process noise for the constant parameters, denoted $\bar{w}_2(k)$, allows the filter to change its estimate of these values through time and allows the filter to track the true time varying disturbance. The set of process noise terms $\bar{w}_1(k)$, $\bar{w}_2(k)$, and $\bar{w}_3(k)$ are stacked as defined by the state $\hat{x}$. All noise terms are assumed zero mean, Gaussian, white, and stationary. This representation results in multiplication and division of estimated parameters

through the dynamics function. Specifically, temperature is being multiplied by RC parameters necessitating non-linear estimation techniques.

Measurement noise is specified based on the accuracy of the temperature sensors. Process noise is specified for the temperature states based on the level of zone aggregation used while the RC and disturbance process noise is set to an artificial value greater than zero in order to allow the filter to vary its estimate of these parameters through time. Increasing process noise level for any parameter indicates that the model is not confident of its ability to describe the process evolution of that parameter. Disturbances, which by their nature the model is not explicitly capturing, are biased and vary with time. In order to estimate the disturbances over time, their noise level is set to be non-zero. Because the RC values should be fairly constant, while the disturbance bias may change throughout the course of a day, the noise level for RC parameters should be much smaller than for disturbances.

A minimal set of independent parameters must be specified for filters to enforce the system dynamics during parameter estimation [27]. Over-parameterization causes unidentifiable parameter manifolds or extra degrees of freedom and can result in violation of dynamics constraints and physics laws such as conservation of energy. In machine learning and system identification, indeterminate degrees of freedom can cause overfitting where the model learns the noise instead of the dynamics of interest. In estimation theory, parameter observability requires that the Fisher information matrix be invertible—redundant parameters or over parameterization breaks this observability criterion resulting in an unobservable subspace [28].

For any thermal network without heat flux information, the total number of

unique parameters is one less than the sum of the total number of resistances and capacitances. RC products are estimated together in order to reduce the non-linearity of the estimation problem. However, by estimating RC products, extra parmaters are mathematically introduced into the estimation problem due to the multiplicative combinations.

Two nodes that have no shared conduction or convection are considered independent, and any edge directly connecting them is pruned from the graph to give the simplest representation. Independent ambient nodes such as external temperatures have infinite capacitance in the thermal network. External nodes may have unique update functions depending on the simulation and weather desired for the modeling exercise.

Because diagonal terms in A are linear combinations of the off-diagonal terms, parameter estimation is only performed for off-diagonals. Parameterization of trees, graphs with no cycles, with combined RC products automatically guarantees a minimal representation of the system.

Unfortunately this minimal guarantee does not extend to graphs containing closed cycles. In a graph, each cycle which uses at least one unique edge and passes through no nodes with infinite capacitance may be used to eliminate one redundant RC product from the estimation problem by multiplying and dividing around the loop.

The parameterization process is best understood through illustration. The building in Fig. 3.4 contains 2 internal zones with 1 resistance plus one external zone with 2 resistances connecting to the internal zones. The external zone is assumed to have infinite capacitance and controlled by a weather forcing function.

This model contains 3 resistances and 2 capacitances, less one, gives 4 unique parameters to estimate. By parameterizing the tree as described, four unique RC products are defined $p_1 = R_{12}C_1$, $p_2 = R_{13}C_1$, $p_3 = R_{12}C_2$ and $p_4 = R_{23}C_2$, and two controlled heater coefficients are defined $q_1 = b_1/C_1$ and $q_2 = b_2/C_2$.

3.2.2 UKF Estimation Procedure

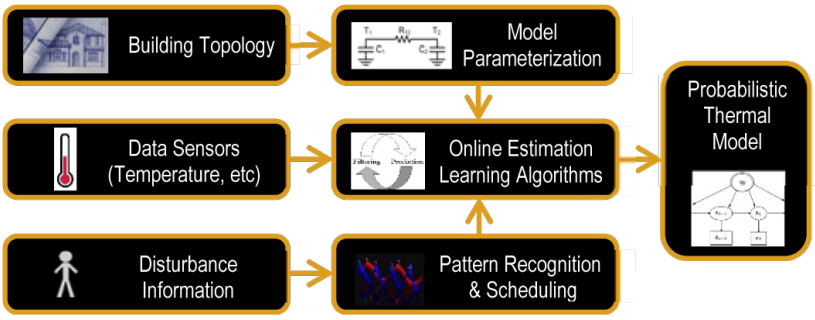

Figure 3.1: Thermal Model Estimation Overview.

Merging data measured from the building, knowledge about disturbances, and first-principles parameterization of the building's topology inside an on-line estimation algorithm provides us with a scalable framework (Fig. 3.1) that can be deployed on buildings. Temperature states are included in the estimation routine because temperature measurements in buildings are not perfect: sensors may be slightly miscalibrated, an air volume may not be thoroughly mixed, and individual thermal zones may have some unobservable temperature gradient. Parameter estimation of the augmented state $\hat{x}$ is achieved in the Kalman Filter. As shown in Equation (3.5), the state dynamics function multiplies parameters $\bar{p}$ by temperatures $\bar{T}$ necessitating non-linear estimation techniques. An Un-scented Kalman Filter (UKF) provides a numerically stable and near-optimal way to estimate parameters, disturbances, and temperature states through the

non-linear Prediction step. The measurement function is linear because each temperature is directly measured, so the regular linear Kalman Filter equations are used for the measurement Update step. For thorough UKF derivation and explanation, please see [58] and [59]. For clarity, the UKF Prediction step and KF Update steps are included below.

The UKF uses the Unscented Transform to pass a distribution through a nonlinear transform. Specifically the UKF samples $(2n + 1)$ points in the distribution, evaluates each point through the non-linear transform and then recombines these points to generate a transformed mean and covariance. The samples, called sigma points, are evenly spaced to capture at least the first and second order moments of the distribution and are weighted such that the covariance and mean of the samples matches that of the original distribution. After being mapped through the non-linear transform the resulting points are multiplied by their assigned weights to determine the transformed mean and covariance. (Note: For brevity in the following Kalman Filter formulations, notation deviates from the modeling section: subscripts denote time rather than node indices.)

Predict:

$$
\begin{aligned}
\lambda &= \alpha^2(L+\kappa) - L \\
\chi_{0,k-1|k-1} &= \hat{x}_{k-1|k-1} \\
W_0^{(m)} &= \tfrac{\lambda}{L+\lambda} \\
W_0^{(c)} &= \tfrac{\lambda}{L+\lambda} + (1 - \alpha^2 + \beta)
\end{aligned}
$$

$$
\begin{aligned}
\text{For: } i &= 1, \cdots, L \\
&\chi_{i,k-1|k-1} = \hat{x}_{k-1|k-1} + (\sqrt{(L+\lambda)P_{k-1|k-1}})_i
\end{aligned}
$$

$$
\begin{aligned}
\text{For: } i &= L+1, \cdots, 2L \\
&\chi_{i,k-1|k-1} = \hat{x}_{k-1|k-1} - (\sqrt{(L+\lambda)P_{k-1|k-1}})_{i-L}
\end{aligned}
$$

$$
\begin{aligned}
\text{For: } i &= 1, \cdots, 2L \\
&W_0^{(m)} = W_0^{(c)} = \tfrac{1}{2(L+\lambda)}
\end{aligned}
$$

$$
\begin{aligned}
\chi^{k|k-1} &= f(\chi^{k-1|k-1}) \\
\hat{x}_{k|k-1} &= \textstyle\sum_{i=0}^{2L} W_i^{(m)} \chi_i^{k|k-1} && \text{State Estimate (\textit{a priori})} \\
P_{k|k-1} &= \textstyle\sum_{i=0}^{2L} W_i^{(c)} [\chi_i^{k|k-1} - \hat{x}_{k|k-1}][\chi_i^{k|k-1} - \hat{x}_{k|k-1}]^\top && \text{State Covariance}
\end{aligned}
$$

Update:

$$
\begin{aligned}
\tilde{y}_k &= z_k - H_k \hat{x}_{k|k-1} && \text{Innovation} \\
S_k &= H_k P_{k|k-1} H_k^\top + R_k && \text{Innov. Covariance} \\
K_k &= P_{k|k-1} H_k^\top S_k^{-1} && \text{Optimal Kalman Gain} \\
\hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k \tilde{y}_k && \text{State Estimate (\textit{a posteriori})} \\
P_{k|k} &= (I - K_k H_k) P_{k|k-1} && \text{State Covariance}
\end{aligned}
\tag{3.6}
$$

For the filter dynamics function $f()$, temperature state dynamics follow the previously derived thermal model in Equation (3.5) while the RC and disturbance parameters are modeled as constants. Artificial process noise for the constant parameters, $\bar{w}_2(k)$ and $\bar{w}_3(k)$, allows the filter to change its estimate of these values through time and allows the filter to track the true time varying disturbance. The set of process noise terms $\bar{w}_1(k)$, $\bar{w}_2(k)$, and $\bar{w}_3(k)$ are stacked as defined by the state $\hat{x}$ and drawn from distribution Q. The measurement noise terms $\bar{v}(k)$ are drawn from distribution R. The standard values of $\alpha = 10^{-3}$, $\kappa = 0, \beta = 2$, typical for a Gaussian distribution, were used to generate the samples χ and weights W. The samples are recombined to give the *a priori* state and covariance estimates. Note that λ and W can be reused so only the χ terms need to be recalculated each iteration. The measurement function of the thermal network is linear as we are directly measuring temperatures, so H is simply an n row identity matrix (provided all temperatures are measured) padded with columns of zeros for the parameters.

The Kalman Filter internally maintains a covariance P_k of the current state estimate $\hat{x}_k$. This covariance is a measure of the accuracy of the state estimate, which equates to the probability density function of the state estimate in a recursive Bayesian estimator. The Kalman Gain K_k is computed based on the information present in the measured outputs. Through the Update step, the Kalman gain will cause the state estimate and covariance to be modified only if the measurement provides new information. In this manner the estimation technique is probabilistic. It is maintaining a thermal model with probabilities attached to each internally estimated parameter and updates estimates and corresponding error probabilities only if measurements provide new information.

Based on the accuracy of sensors R (measured temperatures), accuracy of model Q, and efficiency of estimator, the Cramer-Rao Lower Bound (CRLB) provides a minimum error covariance for the state estimate and is bounded by the inverse of the Fisher information matrix (FIM). The FIM can be thought of as a quantification of the maximum available information about a parameter. As the FIM tends towards infinity the CRLB tends towards zero. In general the CRLB can be written in recursion with the KF based on the information present in all previous measurements, but for a non-linear system a closed form explicit solution is not available [28]. The FIM can be written in an analogous form to the Observability Grammian [60] which provides an alternative perspective—are the system's internal parameters observable given the available measurements. The following section derives an observability matrix which can determine the model parameters that information is available for in a given measurement. Given the system state, the observability matrix correlates information in the current measurement to the respective model parameters.

In order to build an observability matrix, a linear approximation of the system parameter's observability, discrete-time F and C matrices were formed using the true values of the state x and parameters p at the current time step k. The state space F matrix is built as a state transition matrix by calculating the matrix exponential of the Jacobian of the dynamics for the augmented state and

parameters vector.

$$
\begin{aligned}
F(k) &= \Phi(k+1,k) \\
&= \exp\left(\frac{dA}{dx}\bigg|_{x(k)}\right) \\
\frac{\partial A}{\partial x} &= \begin{bmatrix}
\frac{\partial f_1}{\partial T_1} & \cdots & \frac{\partial f_1}{\partial T_n} & \frac{\partial f_1}{\partial p_1} & \cdots & \frac{\partial f_1}{\partial p_m} & \frac{\partial f_1}{\partial q_1} & \cdots & \frac{\partial f_1}{\partial q_l} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\frac{\partial f_{n+m+l}}{\partial T_1} & \cdots & \frac{\partial f_{n+m+l}}{\partial T_n} & \frac{\partial f_{n+m+l}}{\partial p_1} & \cdots & \frac{\partial f_{n+m+l}}{\partial p_m} & \frac{\partial f_{n+m+l}}{\partial q_1} & \cdots & \frac{\partial f_{n+m+l}}{\partial q_l}
\end{bmatrix}
\end{aligned} \tag{3.7}
$$

The state space C matrix is simply an identity matrix for the measured temperature states padded with zeros.

$$
C = \begin{bmatrix} I_{n\times n} & 0_{n\times m} & 0_{n\times l} \end{bmatrix} \tag{3.8}
$$

A discrete time observability matrix was built using the standard form

$$
O(k) = \begin{bmatrix} C & CF(k)) & \cdots & CF(k)^{n+m+l-1} \end{bmatrix}. \tag{3.9}
$$

By examining the rank of $O(k)$ one can determine how many parameters are simultaneously observable. Numerically rank is computed by determining linear independence of columns in the matrix and uses a small value ϵ to as a minimum threshold for the magnitude of the cross product of column vectors. Thus rank returns an integer number. If the rank of the observability matrix is less than the total number of states, then the system is not fully observable. The directions of the column space (also known as image or range) define which parameters information is available for in the measurement.By examining the nullspace vectors, one can determine which parameters are not observable.

In the presented KF formulation, the observability matrix, and by association the FIM, is dependent on the true state x that evolves through time. Intuitively this makes sense, if all temperatures are co-linear, no information about

heat transfer between zones can be inferred, but if at a later time, the temperatures are perturbed, information about heat transfer can be inferred. Mathematically this means that without sufficient excitation, the parameter error lies in the null-space of the auto-covariance [35, Theorem 5.2.5]. Maximizing Fisher Information is one key metric for optimal experiment design and will be revisited in the excitation formulation section. The Kalman Filter maintains estimate covariances through time and updates estimates when measurements contain new information. One interesting question about excitation will be: is a building system fully time-invariant observable when excitation is applied. That is, is information provided about all states when excitation is present, or is the system simply time-varying observable such that an estimator with memory can explore the system parameters through targeted experiments and over time create an overall model of the system.

The probabilistic thermal model which includes parameter estimates and associated error covariances provides additional control options. MPC uses the current state and parameter estimates along with occupancy and weather forecasts to anticipate future disturbances and usage when computing control efforts. The parameter estimate error covariances could be leveraged in a robust MPC implementation that uses chance constraints to bound temperature excursions based on the current estimated model fidelity [61]. For example, particles could be sampled from the model's parameter distributions such that the expected value matches the current parameter estimate. Simulating these particles forward through time and conservatively selecting control efforts that satisfy all particles provides a robust MPC solution that leverages the probabilistic thermal model. Additionally, scenario-based MPC provides a less conservative alternative for trajectory planning of control inputs that also could leverage the

probabilistic thermal model [62].

3.3 Controller Architecture

3.3.1 Baseline Thermostat Control

To evaluate the performance of model-based control, a baseline thermostat controller, similar to what is widely deployed in most current buildings, was developed for the simulation. A winter office building heating scenario was chosen for the simulation study. Minimum and maximum temperature set points were chosen for occupied and unoccupied building zones and tied to a schedule which the controllers could use.

Additionally a turn-on preheat time schedule was added to the temperature profile for the thermostat controller. This preheat buffer time was sized to match the time the heater would take to heat the building from its unoccupied temperature set point to its occupied temperature set point if the outside temperature was 32 degrees. Additionally a hysteresis timer of 15 minutes between turn-on/turn-off events was included to prevent the controller from chattering on and off at every integration time step. Alternatively a Proportional-Integral controller could have been used, but would have required the same preheat buffer and performed similarly to the thermostat for our simple 2-zone building.

3.3.2 Model Predictive Controller

A model-based control method is presented to demonstrate the efficacy of an improved system thermal model. A Model Predictive Controller (MPC) computes an optimal control input trajectory over time to minimize some cost function over a specified horizon given state and dynamics constraints. MPC operates by applying only the current time step control effort and then recomputing the optimal control effort trajectory at the next time step and repeating. Utilizing occupancy, weather forecasts, and variable time-of-usage energy costs, the MPC algorithm can anticipate future disturbances to use minimal energy or incur minimal monetary expense while preserving occupant comfort. Because the control trajectory is recomputed at each time step, the controller is robust to unmodeled disturbances and weather forecast inaccuracies. The following section formulates the optimization function solved at each time step in our MPC implementation.

The formulations given here follow standard convention as presented in [63]. The MPC developed will minimize energy use, saving money, and bound occupant discomfort by ensuring temperatures stay within comfort bounds. The following section starts with a simple quadratic cost and then adds in time-varying temperature bounds for occupant comfort. Energy cost is assumed uniform; time of use charges could readily be added for industrial users.

Dynamics constraints for this problem are based on the thermal model estimated by the UKF. External weather, given at time step intervals by the most recently available prediction, is considered a constrained state and thus included as an additive term T_{ext}in order to minimize the state dimension. An execution

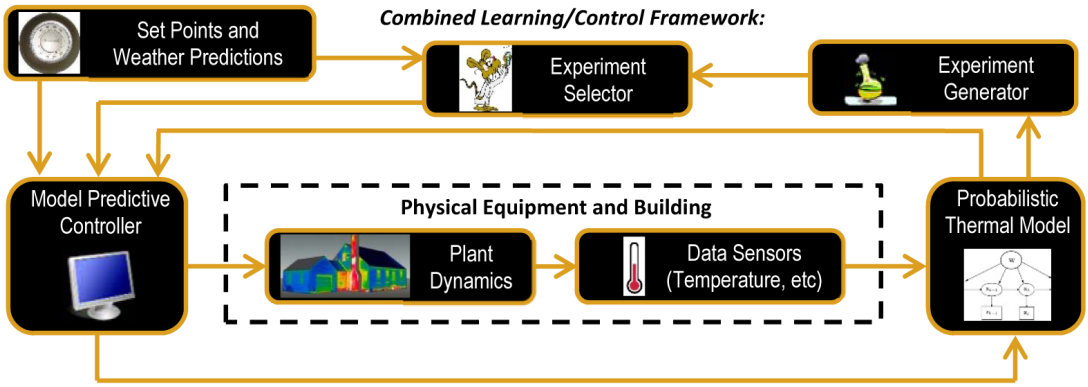

Figure 3.2: Combined framework for parameter estimation, control, and self-excitation.

time step interval of 15 minutes was selected. The discrete state space system is

$$T(k+1) = AT(k) + B_{\text{ext}}T_{\text{ext}}(k) + B_{\text{ctrl}}u(k) \tag{3.10}$$

where $T(k)$ is the temperature vector, $u(k)$ is the control input, A and B_{ext} are populated with RC parameters p_k, and B_{ctrl} is populated with heat coefficients q_l that are learned by the UKF. All control inputs $u_i(k)$ are subject to the constraint set $0 \leq u_i(k) \leq 1$, and $T(0)$ is set to the current temperature.

Equation (3.11) gives an example cost function J to be minimized over horizon h given n controllable temperature states. It is composed of a reference quadratic (2-norm) tracking term multiplied by weight Q plus a 1-norm control

effort term multiplied by weight R.

$$\begin{aligned}
&\min_{u(1)\cdots u(h)} J \\
&J = Q\sqrt{\frac{1}{nh}\sum_{k=1}^{h}(T(k)-r(k))^{\top}(T(k)-r(k))} + R\sum_{k=0}^{h-1}|u(k)| \\
&\text{Subject to:} \\
&T(k+1) = \Phi_A T(k) + \Gamma_{B_{\mathtt{ext}}} T_{\mathtt{ext}}(k) + \Gamma_{B_{\mathtt{ctrl}}} u(k) \\
&0 \le u_i(k) \le 1 \\
&\text{Given numerical values for vectors:} \\
&T_{\mathtt{ext}}, T(0), r
\end{aligned} \tag{3.11}$$

Utilizing the given constraints and cost function, the MPC problem can be posed as a convex optimization problem and solved in Matlab using CVX [64], [65]. Experimentation showed the Gurobi solver [66], available as third-party solver supported by CVX, to be 2 to 10 times faster at solving the MPC problem than CVX's pre-configured SeDuMi or SDPT3 solvers.

3.3.3 Soft Constraints

The cost function given in Equation (3.11) works well for pure reference tracking but poorly for typical energy saving climate control where one only desires that the temperature be kept between two bounds. References [63] and [67] present an alternative method utilizing soft-constraints. Hard constraints, such as keeping the temperature between upper and lower bounds, do not work well for dynamical system outputs because they can make the problem unsolvable under normal operational conditions. Consider if a thermal disturbance pushed the temperature far enough outside of the hard bounds that the temperature

bounds could not be met at the next time step. In such a scenario the solver would fail to calculate an output. Simply speaking, the maximal or minimal control effort applied for one time step would fail to allow the system to satisfy the hard temperature constraints, and thus the controller would do nothing. By using soft constraints the controller is robust to this failure condition. If too far outside the bounds, the controller would simply apply maximal hard-constrained control effort until the temperature is within the soft-constraint bounds.

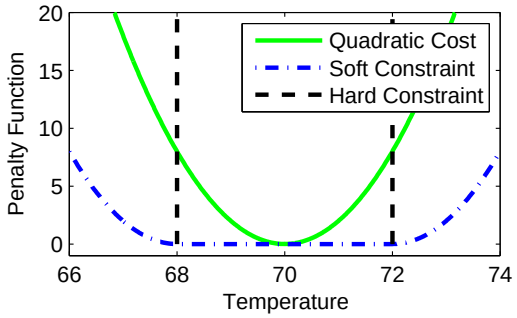

Figure 3.3: Comparison of temperature cost functions.

A comparison of quadratic (2-norm regulating 70°), soft-constraint (2-norm regulating between 68°and 72°) and hard-constraint (bounded between 68°and 72°) penalty functions are shown in Fig. 3.3. A naive method of mathematically writing the soft-constraint might be given by $J_Q = \|\min(T(k) - T_{\min}, 0)\|^2 + \|\max(T(k) - T_{\max}, 0)\|^2$ and preclude the use of any convex solver due to the minimum and maximum functions. By introducing an $n \times h$ size vector of slack variables w, the soft-constraint can be fully formulated in a convex cost function as shown in Equation (3.12). Note that $T(k)$ is vector of length n so T, w, $r_{\min}$, $r_{\max}$, and u are all size $n \times h$. An additional cost to go term was included in J where $r(k) = (r_{\min}(k) + r_{\max}(k))/2$. Given the discrete nature of the problem coupled with control constraints, the cost to go term helps the temperature to tend toward the

middle of the bounded range.

$$\min_{u(1)\cdots u(h),w(1)\cdots w(h)} J$$

$$J = Q\sqrt{\frac{1}{nh}\sum_{k=1}^{h} w(k)^\top w(k)} + R\sum_{k=0}^{h-1}|u(k)| + Q_{\texttt{to go}}\sqrt{\frac{1}{n}(T(h)-r(h))^\top(T(h)-r(h))}$$

Subject to:

$$T(k+1) = \Phi_A T(k) + \Gamma_{B_{\texttt{ext}}} T_{\texttt{ext}}(k) + \Gamma_{B_{\texttt{ctrl}}} u(k)$$

$$0 \le u_i(k) \le 1$$

$$T(k) + w(k) \ge r_{\texttt{min}}$$

$$T(k) - w(k) \le r_{\texttt{max}}$$

$$w \ge 0$$

Given numerical values for vectors:

$$T_{\texttt{ext}}, T(0), r_{\texttt{max}}, r_{\texttt{min}}, r$$

(3.12)

This minimization routine is solved at each time step in the MPC algorithm. The control calculated for the first timestep, $u(1)$, is used. If computation is not real-time, longer time steps or the control effort calculated over multiple time steps $u(1)\cdots u(i)$ can be applied, where i is greater than the time required to calculate the next iteration of the controller minimization routine.

3.4 Matlab Simulated Test Building

3.4.1 Two-Zone Model

Most buildings do not have uniform usage. Office buildings, schools, and stores are generally occupied during the work-day while apartments and houses are generally occupied on nights and weekends. This non-uniform usage in the presence of varying external temperatures is what allows predictive controllers to outperform thermostat controllers—given a weather prediction and the current state, they can better anticipate how the building should be controlled. Our heating control example reflects the typical usage of a heated 2-zone office building in winter that is occupied during the day and vacant at night. The 2-zone building presented here was originally developed in [2].

Fig. 3.4 shows a diagram of the RC network representing the building while Table 3.1 contains the numeric values used in the main MPC versus thermostat simulation study. Values were chosen to correlate with typical cooling and heating time constants for a building measured in degrees Fahrenheit. Notice how zones 1 and 2 are weakly connected: R_{12} has a higher thermal resistance than either R_{13} or R_{23}. This may seem odd, but in fact is a realistic common occurrence, especially for buildings that have additions. The zones might share one wall with each other while each having 3 external walls plus a roof causing easier heat exchange inside to outside than from zone to zone in the building. Using the parameterization previously described, the UKF estimates 4 RC parameters, 2 additive heat terms, and 2 internal temperatures and has measurements of the 2 internal temperatures, the external temperature, and the internal heater control effort.

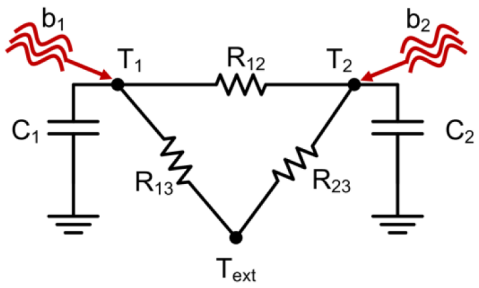

Figure 3.4: Two-zone RC network plus heaters b_1, b_2.

Table 3.1: Two-Zone Model Parameters

Variable	Quantity	Numeric Value	Units
$T_1(0)$	Zone 1 initial temperature	70	°F
$T_2(0)$	Zone 2 initial temperature	70	°F
$T_{\mathtt{ext}}(0)$	External zone initial temperature	20	°F
C_1	Zone 1 capacitance	17	10^5 joule/°F
C_2	Zone 2 capacitance	10	10^5 joule/°F
R_{12}	Zone 1/2 resistance	150	10^{-5} °F/watt
R_{13}	Zone 1/3 resistance	60	10^{-5} °F/watt
R_{23}	Zone 2/3 resistance	100	10^{-5} °F/watt
b_1	Zone 1 heater output	0.18	10^5 watts
b_2	Zone 2 heater output	0.22	10^5 watts

Weather was designed with controlled frequency content to provide an upper bound, high level of external excitation; if the synthesized weather does not sufficiently excite the building to provide information to estimate all parameters in the building, then actual weather certainly will not sufficiently excite the building. Synthetic weather was formulated as the sum of two sinusoids plus an occasional temperature bias and additive random noise. The main sinusoid,

24 hour period and 20 degree peak to peak amplitude, provided daily fluctuation while the secondary sinusoid, 4 hour period and 5 degree peak to peak amplitude, provided slight variations throughout the day. Occasional temperature biasing mimicked hot-front and cold-front temperature swings. In order to demonstrate that the predictive controller was robust to non-perfect weather forecasts, MPC weather predictions were made from the daily sinusoid and bias but not the 4 hour sinusoid or the additive random noise.

3.4.2 Unobservable Parameters

To better illustrate the reliability issues mentioned in the introduction and motivate the self-excitation and monitoring routines, examples of two failed parameter estimation attempts from particularly bad data sets were included (Fig. 3.5). Both examples were conducted by learning four RC parameters of a 2-zone model over a 3 day period: passive dynamics for the first day and active heating for remaining two days. Fig. 3.5.A, based on model in Table 3.1, shows one parameter whose covariance shrunk when heating commenced but whose estimate did not converge on the correct value. Fig. 3.5.B, based on a different model with small thermal resistance between zone 1 and 2, shows a parameter whose estimate became negative when heating commenced. These poor estimates are not an artifact of initial parameter seeding: notice how initializing the parameter value to be larger than the correct value did not prevent the estimator from guessing a negative value. Correcting such problems requires a better excitation signal in the measured data. In the proposed framework, the monitoring routine would detect a negative parameter or run a bank of filters to detect a lack of consensus on a parameter estimate and with self-excitation

attempt to converge on a better parameter estimate.

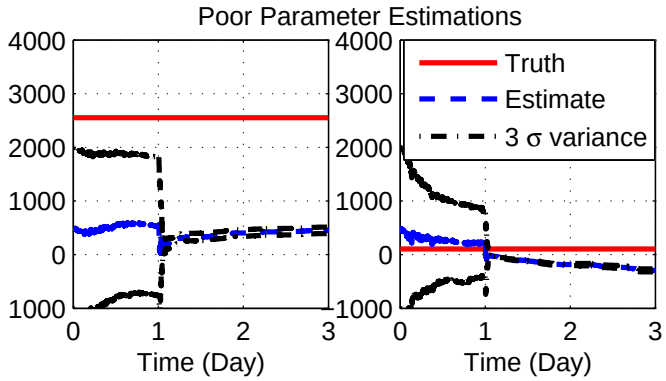

Figure 3.5: Poor parameter estimation with UKF. Initial passive learning for the first day, heating commences at end of day one. Parameters estimated are RC products which have units of seconds. A.) (left) Parameter covariance shrinks with poor estimate, B.) (right) Parameter estimate becomes negative.

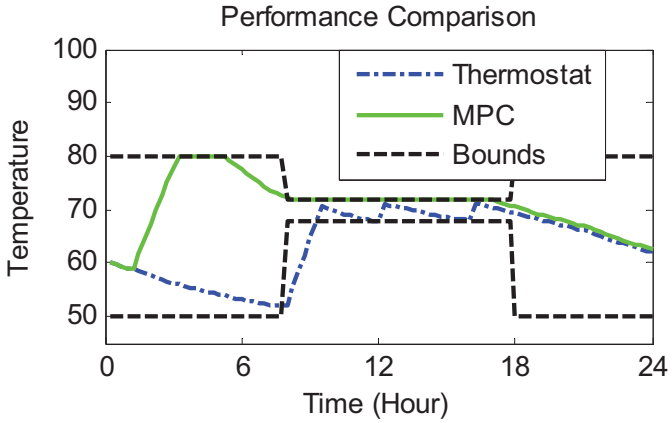

Figure 3.6: MPC errant behavior is due to bad model parameters.

Thermal models with very bad parameter estimates cannot be reliably used for MPC. Using the parameters poorly estimated for the model in Table 3.1, a simulation was run comparing MPC performance against a baseline thermostat. MPC controlled zone 1 well but control of zone 2 was errant, see Fig. 3.6. The model contained bad parameters for the edge connecting zone 1 and zone 2, which caused MPC to attempt to apply extra heat to zone 1 by heat-

ing zone 2 to its upper temperature bound. MPC's poor performance, despite obeying temperature bounds, caused 30% more energy to be used compared to the well-tuned thermostat controller. Results such as these necessitate both self-excitation and monitoring routines to be part of the online estimation and control framework.

3.5 Self-Excitation and Monitoring

3.5.1 Overall Framework

Proper operation of MPC is dependent upon a good thermal model of the building. It is also widely recognized, however, that obtaining the process model is the single most time consuming task in the application of model-based control [68]. Due to the inherent low level of excitation typically present in most buildings, active learning is an essential component to a robust online system identification routine [32]. Another perspective to view the problem is as a need to weed out candidate models. That a scientific theory may be falsified by contradicting evidence but never validated by corroborating evidence was elaborated on by the philosopher Karl Popper [69]. The information contents in the observed data corresponds exactly to the set of models that are falsified by the observed data. The more informative data is, the larger is the set of candidate models that can be falsified [70]. In summary the experiment should be as short as possible and should disrupt the process minimally. In machine learning this is a classic problem tradeoff between exploration and exploitation, should the current estimate be exploited for currently best-known control, or should

the current estimate be improved via exploration such that the improved estimate can be exploited for future improved control. The overall problem can be viewed from a Design of Experiments (DOE) perspective where we wish to design optimal or efficient experiments that extract the most new information possible while consuming the least energy, minimally disrupting occupants, and not violating any physical hardware constraints [57]. Formulations derived in this section explore various designs, all of which obey operational building constraints while providing alternative views of selecting the most relevant new information to extract by self-excitation of the building.

For actively heated and cooled buildings, this active learning can be accomplished by self-exciting the building's thermal dynamics. The goal is to find which system input to excite in order to reduce the uncertainty of a parameter. This can be accomplished by finding which zone is associated with a parameter of interest and exciting the zone's actuator. A combined learning, control, and self-excitation framework has been proposed in Fig. 3.2, which was built from a typical MPC system and augmented with the self-excitation components: Experiment Generator and Selector.

Functionally speaking the Generator provides a listing of parameters that are poorly known in the thermal model and the Selector chooses an appropriate time to actively learn those parameters by exciting the system. Specifically the Experiment Generator uses covariance and model sensitivity criteria to correlate poorly known parameters, back to physical zones that can be excited. Given the building state and forecasted conditions, the Experiment Selector evaluates whether it makes sense to excite any physical zones, at a given time, by modifying temperature set points over a short horizon. The goal is to not significantly

disrupt occupants or waste energy while trying to improve the learned model. The following sections cover these active learning functions.

3.5.2 Experiment Generator

The UKF estimates parameters based on measurements of the building temperatures and updates a covariance matrix for the set of parameters. The covariances can be thought of as an inverse information criterion for the estimated parameter. So a small covariance for a given parameter implies that parameter is known well. In general we assume the UKF covariances accurately represent the quality of the parameter estimate, that is, we assume the estimator is unbiased. The Experiment Generator attempts to increase the Fisher Information available from future measurements by determining which zones should be excited such that weakly estimated parameters may be better estimated.

Recall that estimated parameters correlate with actual resistances and capacitances of physical zones in the building. The Experiment Generator correlates poorly known parameters back to physical zones that can be excited. We propose three methods for identifying zones to excite:

- Covariance Eigen-decomposition
- Variational temperature methods
- Monte Carlo energy methods.

Below is an exploration of each method.

The covariance Eigen-decomposition method attempts to pick the most

uncertain parameter subspaces and then make an ordered list of the zones which correlate to those parameters. The approach is analogous to finding the nullspace directions of the observability matrix and exciting zones such that the range space of the observability matrix spans the directions of largest uncertainty in the parameter space. Exciting the listed zones should provide information to improve the worst parameter estimates shrinking the largest covariances. By computing eigenvalues λ and eigenvectors ν of the subset of the covariance matrix $P_{(k|k)}$ that is associated with RC parameters, the Eigen-decomposition in Equation (3.13) creates a prioritized listing of parameter vectors that the estimator needs help to actively learn through excitation. The listing is ordered by largest singular-value which correlates with the most uncertain parameter set[1].

$$[\lambda, \nu] = \text{eig}(P_{RC\ k|k}) \tag{3.13}$$

To determine the appropriate excitation, this listing of uncertain parameters must be correlated back to physical thermal zones as one cannot directly excite parameters. Thus the eigenvector ν_m for each eigenvalue λ_m must be related back to the respective element of A based on how A was constructed in Equation (3.4). Reciprocals of estimated RC parameters were used to build the A matrix, but a parameter sensitivity analysis of this matrix would show that the primary factor of interest is solely which nodes have connected edges with large variances. Using Equation (3.4) a simple correlation function in Equation (3.14) has been created to define temperature node uncertainty weight vector N_m for each eigenvalue λ_m. External nodes have infinite capacitance, so no parameters are estimated for these nodes. This causes the row in A_{λ_m} corresponding to any external node to have all zero elements. In order to represent excitation relative to an external node i, the ith column is summed and stored in the (i, i) term

[1]Singular values for a covariance matrix equals the absolute value of the eigenvalues.

of A_{λ_m}. The vector N_m contains large values for nodes whose edge-connected parameters are poorly known. Here index k correlates with parameter $C_i R_{ij}$ as defined by element p_k in the vector $\bar{p}$ minimal parameterization. For example, in the simulation results parameter p_1 corresponds to $R_{12}C_1$, p_2 to $R_{13}C_1$, p_3 to $R_{12}C_2$ and p_4 to $R_{23}C_2$. If any redundant parameters were multiplicatively canceled out in the minimal parameterization step, then their multiplication relation will need to be substituted back into the appropriate edge in Equation (3.14) by performing the same multiplication on the Eigen-vector components.

$$
\begin{aligned}
A_{\lambda_m} &= \begin{cases} A_{ij} = 0, & \text{if } i \neq j, (i,j) \notin E \\ A_{ij} = \nu_{m,k}, & \text{if } i \neq j, \in E \\ A_{ij} = \sum_{i \neq l} A_{il} - \sum_{i \neq l} A_{li}, & \text{if } i = j \end{cases} \\
N_m &= \text{diag}(A_{\lambda_m})
\end{aligned}
\tag{3.14}
$$

By analyzing vector N_m we can see which nodes should be excited relative to each other. This information, the set of vectors N and eigenvalues λ, is passed to the Experiment Selector which can actually create excitation by modifying temperature set points accordingly to create a difference in the temperature between nodes.

If the RC model was heavily biased, the covariance for the estimated parameter may be an unreliable metric of estimate quality. Additionally, all parameters may not significantly effect the temperature estimates of all zones in the building. Thus in certain situations the priority in excitation might be better posed by examining the sensitivity of zone temperatures to individual parameters. We call this a variational method which can be mathematically posed as a partial derivative of temperature with respect to parameters as the Jacobian of

the state space system

$$\frac{\partial A}{\partial \bar{p}} = \begin{bmatrix} \frac{\partial T_1}{\partial p_1} & \cdots & \frac{\partial T_1}{\partial p_k} \\ \vdots & \ddots & \vdots \\ \frac{\partial T_i}{\partial p_1} & \cdots & \frac{\partial T_i}{\partial p_k} \end{bmatrix} \tag{3.15}$$

Alternatively the variational technique could be posed with a sigma-point method as described in [71] for optimal experiment design.

Parameters are also being estimated for the HVAC components, and the controller is using both the thermal dynamics and actuator constraints when selecting a control action. The energy consumption across zones may be imbalanced, so selecting the parameter that affects the most temperatures may not result in initially saving the most energy. Monte Carlo methods could be used to examine the sensitivity of energy consumption from the controller to parameter variations in the thermal model. By sampling parameter values, a Monte Carlo simulation could be run to examine $\partial E/\partial \bar{p}$, the partial derivative of energy with respect to the parameter values.

The parameter variances that most influence energy usage should be estimated best through system excitation. The main limitation to the variational and Monte Carlo methods is that they are being run on the simulated model. Given a bad initial model this method will not give meaningful insight. We recommend the Eigen-decomposition method for buildings with no strong prior model (e.g. any building where one expects to use a gray-box online learning method to acquire an initial model.) We recommend the variational method or Monte Carlo method for buildings which have a strong model, either after initial model convergence or from a strong Bayesian prior. These methods are applicable when one wishes to continue to refine their parameter estimates over time to capture the building's change in use, wear/aging, fault detection, or general

purpose monitoring.

3.5.3 Experiment Selector

The Experiment Selector evaluates the query: at the given time does it make sense to modify any temperature set points over a horizon h given current building state, occupancy, upcoming temperature bounds, weather predictions, and current thermal dynamics model? During initial acquisition without a Bayesian prior, one main limitation is that the large uncertainty of the current estimated model may cause the dynamics and baseline control calculations to be erroneous. If the UKF has not converged on parameter estimates then the MPC control will be unreliable and any routine that uses the estimates is liable to produce erroneous behavior—thus, until parameters have converged, thermostat control should be used. For new buildings an initial model is often available from the designers to seed the UKF. For retrofit buildings data may be collected until model convergence before running MPC. In any case, properly introducing excitation into the system reduces this learning time.

In software simulation it is common to see pseudo-random binary inputs used for excitation to generate control-oriented models from high fidelity plant simulations. While this works great in simulation, pseudo-random binary excitation signals subject chiller, heater, pump, or fan hardware to unnecessary wear and tear. One could create bounded pseudo-random binary inputs for specific components that have been examined to ensure they can handle the input spectrum [49]. The cost of energy savings over a month or even year timeframe is often significantly less than the replacement cost for one broken component.

That is, the risk is too high to use wide bandpass excitation signals, therefore we seek strategically targeted excitation signals. Keeping this in mind we present two experiments, one simple heuristic for perturbing temperature values and a model predictive option.

The heuristic options looks at the first vector N_m and picks out the two largest values—let i and j contain their respective node ids. If the values are within 25% of each other, then: case 1) excitation is desired between node i and j; otherwise: case 2) excitation of node i relative to all other nodes is desired. Given the current temperatures, predicted weather, and temperature bounds, determine if heating (or cooling) the desired nodes will cause a noticeable increase in the temperature difference of interest. If so, then apply heat for four subsequent time steps or until an upper (or lower) temperature bound is reached.

After initial acquisition, when MPC is running, the decision and temperature modification can be formalized as a convex optimization problem using the format developed for MPC. To seed the problem, results from the current MPC solver iteration provide baseline temperature differences and baseline control efforts for the original temperature bounds. The optimization problem is configured to maximize the temperature difference of interest by varying individual zone temperature bounds e, while observing thermal dynamics, actuation saturation limits, hard constraints on the original temperature bounds, and a threshold (e.g. 110%) control effort compared to the unexcited scenario. Mathematically this can be written as Equation (3.16).

$$\max_{u(1)\cdots u(h), e(1)\cdots e(h)} J$$

$$J = \frac{1}{n}\sum_{k=1}^{h} |T_i(k) - T_j(k)|$$

where i and j are nodes of interest

Subject to:

$$\begin{aligned}
T(k+1) &= AT(k) + B_{\texttt{ext}}T_{\texttt{ext}}(k) + B_{\texttt{ctrl}}u(k) \\
0 &\leq u_i(k) \leq 1 \\
T(k) &\geq r_{\texttt{min}} \\
T(k) &\leq r_{\texttt{max}} \\
\sum_{k=1}^{h-1} u(k) &\leq 1.1 \times u_{\texttt{baseline}} \\
e(k) &\geq r_{\texttt{min}}(k) \\
e(k) &\leq r_{\texttt{max}}(k) \\
T(k) &\geq e(k)
\end{aligned} \tag{3.16}$$

Given numerical values for vectors:

$$T_{\texttt{ext}}, T(0), r_{\texttt{max}}, r_{\texttt{min}}, u_{\texttt{baseline}}, i, j$$

If the excited temperature difference J minus the baseline temperature difference is above a threshold then the experiment should be run. This selection routine is then repeated for each smaller eigenvalue. If no selection is made then the threshold is slightly decreased for the next timestep iteration in order to promote occasional excitation experiments. When a selection occurs, then the excitation is run until the horizon h expires and the threshold is reset.

3.5.4 Monitoring

We showed in [1] that parameter estimates were robust to solar radiation and ground coupling disturbances, however, while performing this paper's study, heating loads were observed to occasionally degrade the stability of RC parameter estimates. Two problems occasionally arose: physics violations and false convergence. Physics violations occur when an RC parameter is estimated to be negative, which violates conservation of energy. False convergence occurs when the covariance errantly shrinks while the filter is doing a poor job of estimating a parameter. Shrunken covariance can prevent future data from correcting a parameter estimate. Detecting the former is easier than the latter, but both can be ameliorated with an auxiliary monitoring routine. In this study a human operator monitored performance.

While an estimator is running, the monitoring routine occasionally saves the current parameter estimates and covariances. By simply stopping estimation when a parameter gets close to zero or becomes negative, the estimator can be re-initialized to the last saved checkpoint. This allows poor data sections to be thwarted from corrupting estimates.

False convergence can be combatted with consensus testing. Every so often a new copy of the UKF can be initialized with different initial parameter seeds and run in parallel with the original UKF filter(s). Parameter values are compared based on their expected value and covariance—how many standard deviations away are the other filter's estimates of that parameter. Typically a consensus filter can increase the robustness of initial acquisition routines to poor initial data and poor initial seed estimates. Correcting these numerical issues was outside the scope of this study.

3.6 Matlab Simulated Building Results

3.6.1 Excitation and MPC Performance

Running the same 3 day simulation on the model from Table 3.1, with the addition of excitation, significantly improved parameter estimates. Fig. 3.7 shows the improved estimation of one RC parameter. Notice when excitation was applied during the third day, the parameter estimate is improved to match the true value and the covariance shrinks.

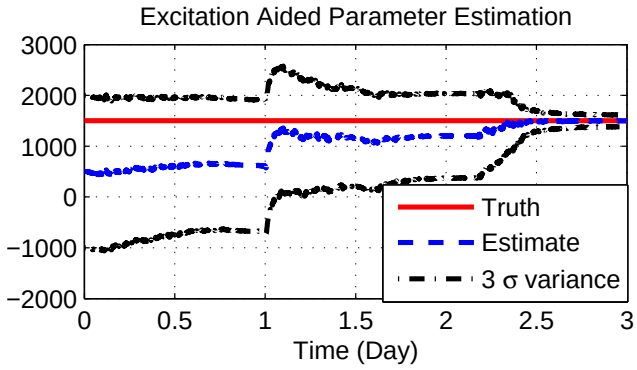

Figure 3.7: Excitation aided parameter estimation with an UKF of an RC product, which has units of seconds. Significant estimation improvement is observed due to excitation. Day 1 passive thermal response, day 2 uniformly heated building, day 3 excitation applied.

Table 3.2: MPC Versus Thermostat Control

Controller	Occupant Discomfort	Energy Usage
Thermostat	0.170	415
MPC	0.086	384

A weeklong control simulation comparison was run between the well-

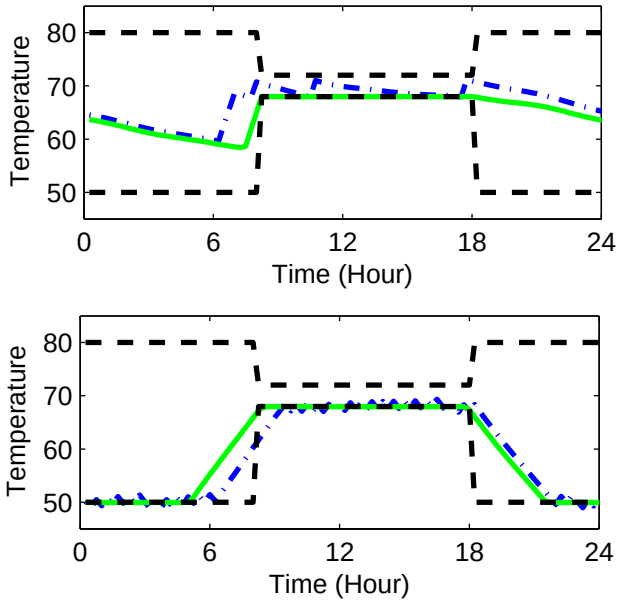

Figure 3.8: MPC performance from excitation aided model estimation. A.) (top) Warm external temp, thermostat heating too early, B.) (bottom) Cold external temp, thermostat heating too late. See legend from Fig. 3.6.

tuned thermostat controller and MPC using the model acquired after excitation. Weather was varied throughout the week from averaging in the mid-fifties Fahrenheit for the first couple days to the teens for the last couple days. Table 3.2 contains the numeric results of the simulation. The MPC used 7.5% less energy than the thermostat controller while improving comfort, defined as the temperature error term from Equation (3.12), by almost 50%. In this 2-zone system it was easy to observe in Fig. 3.8 where MPC outperformed the thermostat: on cold days the thermostat would turn on too late, on warm days the thermostat would turn on too early.

3.6.2 Analysis of Excitation Effects

An observability analysis was performed to provide insight into the mechanics of the estimation algorithm's performance during excitation. The true RC parameters for the system are known and constant but the temperature states vary through time. For the 2-zone house there are 3 temperatures, 4 unique RC parameters, and 2 disturbance parameters for the heaters. Because disturbances were estimated accurately and only learned while the heating system was on, they were omitted from this analysis. This study focuses on the observability of the RC building dynamics.

For the entire simulation the matrix had a maximum rank of 5, meaning 2 states were unobservable in any individual temperature configuration. By looking more closely at the components of the two nullspace vectors v_1, v_2 of the observability matrix, we observe that as the excitation causes zone temperatures to vary, the primary axis directions of the nullspace change as can be seen in Fig. 3.9. Specifically we observe that the excitation applied on the third day causes a significant increase in information about parameters $p_1 = R_{12}C_1$ and $p_3 = R_{23}C_2$ and explains the estimate improvement for those parameters.

The excitation did not improve the number of parameters that can be simultaneously estimated at one instant, but rather changed which parameters we could estimate at a given instant. Although the entire building may not be fully observable at any individual time step, over time, by retaining prior learned information, the entire building's model may be inferred. In other application areas, such as magnetometer based spacecraft attitude estimation, the system's dynamical evolution coupled with a Bayesian estimator is often used to fully observe systems that were initially thought to be partially observable [72]. The

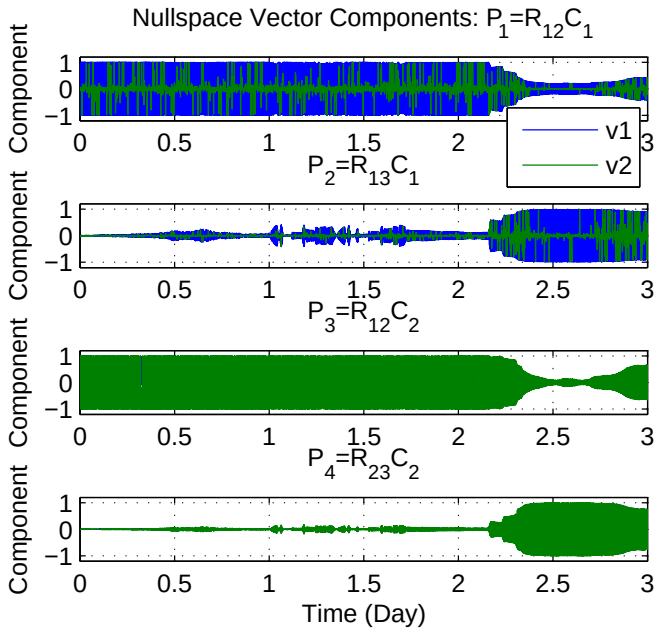

Figure 3.9: Nullspace component magnitudes over time.

Kalman Filter is fusing information when it is available to improve and maintain estimates of all parameters over time. In summary the building system is not time-invariant observable, but it is time-varying observable given appropriate system excitation using the described Experiment Generator and Selector.

3.7 Discussion

The main contribution of this paper was the presentation of methods for self-excitation of the HVAC system of a building. A simple thermostat controlled 2-zone building needed excitation in order to learn a gray-box thermal model given no prior parameter estimates. One of the more insightful observations was finding that excitation did not increase the total number of instantaneously observable parameters in the system but rather changed which specific param-

eters inference could be drawn on from the data. In practice the amount of natural excitation from external disturbances and the HVAC system will vary across buildings. Given that the simple 2-zone model presented here required excitation, was not time-invariant observable, and in agreeance with a number of studies done with real-data, we believe most buildings will require some form of excitation to learn control-oriented models. More complex buildings are likely to be in even greater need of excitation for proper identification. The specific implementation details of the thermal model estimation or predictive controller could be modified and refined, but there will always be a need for useful excitation and robust monitoring if such a system will ever be deployed across real buildings.

The Experiment Generator and Experiment Selector methods presented in this paper intuitively capture the general intent of self-excitation for buildings and worked for initial acquisition of the 2-zone building presented here. The next steps would be to evaluate their performance and computational feasibility in a co-simulation study where MLE+ links Energy Plus to Matlab. Most co-simulation model-inference studies assume infinite hardware response from the simulation and use pseudo-random binary inputs. Running the self-excitation in software while obeying real physical hardware constraints would enable tuning of the various self-excitation routines. Lastly this self-excitation routine will need to be tested on real hardware in real buildings.

The simulation conducted in this study demonstrated the robustness MPC building control has to weather uncertainty, while utilizing a dynamical model learned from an online gray-box estimation method. MPC demonstrated better occupant comfort and reduced energy use versus thermostat control.

3.8 Conclusion

For a 2-zone building a technician may be able to tune the thermostat for a couple common scenarios, but in more complex buildings with more zones, disturbances, and constraints, operator tuning becomes infeasible. Our simulation results, where MPC used an online learned model to improve occupant comfort and reduce energy consumption by 7.5% compared to a well-tuned thermostat controller, demonstrate the potential that online, self-excitation active learning methods have in building systems. Various online methods were presented for excitation in a design of experiments framework, and the effects of excitation on the estimation filter were analyzed to demonstrate time-varying observability and improved Fisher information. By coupling gray-box estimation and predictive control algorithms with online self-excitation and monitoring into a single framework, we have highlighted the potential for low-cost scalable methods to save energy and improve occupant comfort in buildings.

CHAPTER 4

ALL WEATHER PERCEPTION: JOINT DATA ASSOCIATION, TRACKING, AND CLASSIFICATION FOR AUTONOMOUS GROUND VEHICLES

4.1 Introduction

The past decade has seen rapid advancement in autonomous ground vehicles (AGV) from academic research projects: five vehicles successfully completed the second DARPA Grand Challenge and six completed the DARPA Urban Challenge; and industry vehicle research: numerous companies from traditional automotive sectors such as General Motors and Volkswagen, military sectors such as Oshkosh Trucks, and information technology sectors such as Google, safely driving hundreds of thousands of miles with autonomous vehicles on public roads; to public policy advancement: Nevada became the first state to license autonomous vehicles and semi-trucks. Both the NHTSA [73] and SAE [74] have published roadmaps for the future development of autonomous vehicles and have posed the Holy Grail of "Level 4 or 5" as full autonomy in any environment and situation. One might exclude unique situations such as a wild African Safari through the unmarked bush, but commercial autonomy fit for public use is generally understood to include standard driving on all modern road systems. A key shortcoming of current work is autonomous driving in all weather conditions. This is precisely where the limits of current advanced driver assistance systems (ADAS) as well as AGV development and testing are.

At the 2014 Future Automobile Technology Competition in South Korea, rain fell the morning of the second day of testing [75]. The result? Two autonomous vehicles that navigated the course successfully on the first day crashed on the

second—predominantly a result of perception system failures solely due to a wet road and humid conditions. The message was clear: autonomous vehicles are not ready for public adoption until they have been validated and tested to work in all-weather scenarios. While it is true that prudence can cause even the most talented drivers to avoid certain weather conditions, realistic public acceptance of fully automated driving capabilities requires robustness to reasonable and common weather and lighting phenomenon such as snow, rain, fog, and night conditions. The aim of this paper is two-fold. First, a set of data logs are collected in varying weather and lighting conditions to evaluate perception algorithms. Second, perception robustness in all-weather conditions is improved by a joint Bayesian solution to association, tracking, and classification that includes state-of-the-art vision algorithms in addition to lidar and radar sensors.

Since the DARPA Challenges, many advances have occurred in computer vision, lidar point cloud segmentation, and vehicle embedded computing—all of which have direct application in autonomous vehicles. The field of vision-based object detection and localization has experienced several notable innovations such as the development of (1) good low-level feature descriptors that capture local shape, but remain invariant to local photometric and geometric changes [76], (2) models that capture larger scale deformations not captured by the low-level features themselves and methods for the discriminative training of these models [77], and (3) deep learning methods that learn a rich hierarchy of low-level to high-level features from data with little to no manual engineering of the model structure [78], [79], [80], [81], and [82]. The field of deep learning and deformable parts models [83] and [84] has advanced from hand-designed feature detection methods to learn hierarchies of features in an unsupervised manner directly from data, vastly improving algorithm detec-

tion and classification accuracy. Lidar point cloud processing methods have improved to enable accurate segmentation and classification of high resolution 3d point clouds in real-time [85] and [86]. Aligning point-clouds with iterated closest point methods has been shown to improve tracking performance of obstacles' absolute ground speed, an inherently noisy parameter when estimated as a derivative of position in a parametric filter [87]. The entire field of GPU processing for parallelizable computation was developed with the introduction of NVIDIA's proprietary CUDA platform and later support by AMD and Nvidia for the Open Computing Language (OpenCL), enabling real-time computation of that which was previously relegated to a cluster or server farm. Further advancements have demonstrated real-time performance of DPM vision detection methods [88] and [89], and hardware advances in embedded computing have shown deep learning classification running on rugged mobile platforms [90].

With all the recent advances, there have also been some patterns highlighting the current opportunities for future development. With a few exceptions, the vast majority of published studies to date have demonstrated the performance of autonomous vehicles in optimal weather conditions, such as sunny daytime. For example, many well published autonomous driving research efforts have focused their testing in the sunny, fair-weather areas of California and Nevada. This motivates the need to understand how sensors and perception algorithms handle changing weather conditions.

Figure 4.1 shows the major components commonly found in AGV systems, known generally as segmentation and clustering (processing raw sensor analog or digital data into obstacle-level meta-measurements), data association (determining which measurement came from which static or dynamic environmental

obstacle), tracking (estimating the obstacle's state, position, velocity, etc.), and classification (distinguishing cars, people, buildings, etc.). [91] stated that full joint solutions to this perception problem are intractable to formulate or compute. Many advanced techniques recently published have evaluated performance of some of these components in isolation from others in the overall perception pipeline [92], [93], [94]. Examples include: improved tracking and classification algorithms that ignored any segmentation or data association errors [91], performed evaluations on very limited types of scenarios such as tracking a large number of stationary cars or a small number of dynamic objects [87], simply lacked access to large public-domain accurately labeled urban data sets available to quantitatively evaluate performance [86], or handled classification separately after combining data association and tracking [95]. These aforementioned studies have plenty of merit, but the lack of a joint solution is both a concern in practical solutions for an AGV and also an opportunity for development. This paper presents a full joint solution implemented in real-time and tested in both real-world urban environments and repeatable staged scenarios for qualitative and quantitative evaluation across different weather conditions.

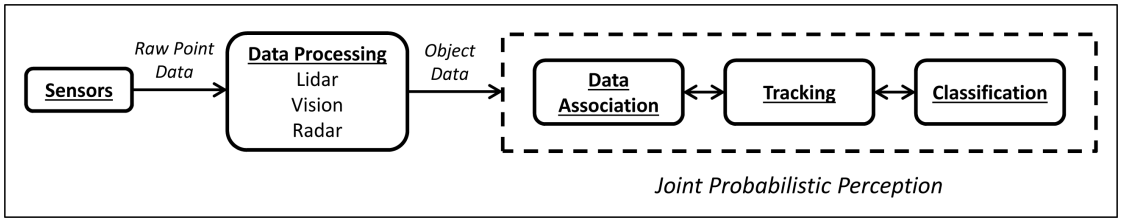

Figure 4.1: Typical AGV perception pipeline.

A corresponding philosophical question arises in robotics literature [96] regarding the importance of actually modeling the dynamics of all possible object classifications if the inputs are truly unknown. In a vehicle, changes in speed

and direction are almost entirely due to driver inputs limited by vehicle performance, which in practice can negate the ability of a kinematics classifier to differentiate object types. For example, a conservative driver may drive a Corvette with the same slow acceleration as an 18-wheeler semi-truck. Despite the difficulty, classification is innate to human driving and perception; the moment before an accident one may attempt to quickly maneuver. Understanding if the object on the side of the road is a human, a shrub, or a boulder is critical to that crash avoidance decision. When driving, a human classifies objects based on their appearance and actions. A camera may detect if an object is a person or a car, but by their motion a person can infer if extra caution must be exercised around the object, such as children playing in the street or a distracted driver. There is utility in Bayesian inference of classification combining sensor detected information and kinematics tracking information. One example approach might be to utilize a bank of filters to detect normal driving versus erratic driving; classification could be passed to a safety-conscious system to leave a larger berth around erratic vehicles. There are some computational concerns which arise, but the number of unique kinematic classes of dynamic roadway objects is small and in practice this inherently helps limit the maximum computational overhead required to support classification.

With the rapid development in the field, the architecture for a proto-typical autonomous ground vehicle is not yet set in terms of sensing, computation hardware, or programming interfaces. Google, for example, heavily utilizes background subtraction for lidar sensor processing, subtracting sensor returns from a pre-built 3D static environmental map to identify moving obstacles [97]. Of any company, Google may have the best infrastructure to 3d map out every public road in the United States, but one can readily point out operational

limitations of such a system to exclude, off-road excursions, private driveways/roads, road-construction re-routing, theater of war, and operation after a natural disaster—a time when one may be most dependent on vehicle mobility. This motivates an interesting question: is *a priori* mapping required or even necessary? Other companies have focused on building systems around different combinations of radar, camera, infared, lidar, and ultrasonic sensors [98] and [99].

Given the current variances in AGV sensing and architecture, and in an effort to understand the challenges adverse weather and lighting scenarios pose, this study examines how the different sensing modalities of radar, lidar, and camera perform in diverse environmental conditions including snow, rain, fog, and nighttime. A particular focus is given to understanding how state of the art vision processing compares to lidar—the predominant sensor of most successful DUC teams. In adverse weather, assumptions about other layers of the perception pipeline can no longer be guaranteed. Furthermore, different layers of the perception pipeline potentially can aid one another—object classification may inform the dynamics of the object to better track it, or existing object tracks may aid data association for region-of-interest image detections. To highlight an example of potential capabilities joint perception solutions provide, a demonstration of object classification based on multiple model Kalman filter tracking is presented for objects in an urban environment. Such multiple model tracking has been shown beneficial for tracking airplanes in turning and straight flight [100] but is novel for classification of terrestrial objects in urban vehicular environments. Kinematics-based classification might be beneficial in precipitation as camera detections or lidar returns become obscured or when tracking solely with radar. The authors believe that one of the keys to handling adverse envi-

ronmental operating conditions is a full Bayesian probabilistic joint perception system, thereby minimizing the number of brittle ad hoc design choices which tend to fail under uncommon untested weather scenarios. This paper utilizes Skynet, Cornell's autonomous 2007 Chevrolet Tahoe from the DUC, and builds upon [95] to extend joint data association and tracking to include classification; relaxations allowing computational feasibility come from a Rao-Blackwellized Particle Filter (RBPF), multiple hypothesis modeling, and carefully managing measurements in forward-pass parametric filters.

In summary this paper makes the following contributions:

- Demonstrates object classification in an urban environment based on multiple model tracking.
- Demonstrates a real-time joint probabilistic method to solve data association, tracking, and classification for an AGV roadway environment.
- Examines if and how state-of-the-art vision algorithms can compliment or replace lidar and radar sensors.
- Investigates sensor and perception algorithm performance in adverse weather and lighting conditions.

4.2 Joint Probabilistic Formulation

Before deriving a full Bayesian formulation for joint data association, tracking, and classification, a brief example is given to demonstrate how measurements fed into a tracker can be used to correctly classify the object solely based on dynamics without any sensor-specific meta-information on the object's shape,

size, color, or type. Kinematics-based classification methods which match an object's dynamics model with measured data points typically require access to measurement update residuals (innovations), and covariances from a Kalman Filter (KF) or Particle Filter (PF). By building upon the brief classification example, a full Bayesian formulation is then developed that extends the combined data association and tracking RBPF from [95].

4.2.1 Joint Classification and Tracking – Derivation and Example

In contemporary literature, direct object classification typically focuses on image processing techniques such as feature extraction or constructing a lidar point cloud and comparing against pre-classified 3d models. Similar work has been done by [101] to combine a KF with extra sensor information such as imagery data to infer object classification. Alternative approaches known as boosting methods have combined banks of weak classifiers to infer object classification; AdaBoost is one of the most popular [102].

Reliable inference of classification is accomplished here by extending a standard Kalman filter tracker given noisy position information of a target, which could be collected from any standard AGV sensor such as radar, camera, or lidar. Seperate classifiers are designed for cars, pedestrians, cyclists, and buses– four of the most common moving objects in an urban environment.

Kalman Filter derivations depend on the inherent uncertainty in the system dynamics and measurements. By assuming Gaussian distribution uncertainty,

the modeled system can be viewed as a mixture or compilation of Gaussian distributions. In a probabilistic graphical models framework, a standard KF is a hidden Markov Chain that has observed noisy measurements where the chain describes the evolution of the dynamic system through time. This chain is illustrated in Figure 4.2, where $\mathbf{x}$ is the hidden system state and $\mathbf{z}$ is the observed measurement. Implementing a KF requires knowledge of the process, dynamics, process noise, and measurement noise. Combining this information along with an estimate of the state initial condition, the KF operates online, updating its estimate or inference of the system state at each time step k, calculating $p(\mathbf{x}_k|\mathbf{z}_k)$.

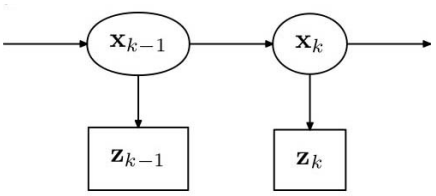

Figure 4.2: Kalman Filter as Hidden Markov Model.

Measurements are assumed to be received from a specified unknown object whose classification exists within a set containing uniquely modeled dynamic processes. As shown in Figure 4.3, the object classification $\mathbf{C}$ is inferred based on the correlation between the dynamic model and the measurements as $p(\mathbf{C}|\mathbf{z}_k)$. Developing a KF requires proper tuning of noise parameters in order to best match the model with the physical system. Innovation test statistics are typically used to validate this matching. The example here uses innovation statistics in a batch methodology to associate the correct dynamic model with the measurement data. Reliably inferring the classification requires accurate and computationally simple models of the dynamic processes.

Pedestrians can walk in any direction while cars, buses, and cyclists are sub-

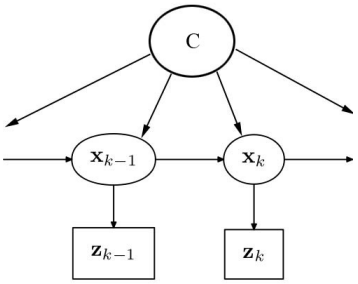

Figure 4.3: HMM dependent on model classification.

ject to non-holonomic constraints of rolling wheels. Inputs are unmodeled so the process noise terms must account for all changes of direction and speed. The dynamics model of the person is assumed to have Gaussian process noise equal in x_1 and x_2 directions given by the following differential equations:

$$\begin{aligned} \ddot{x}_1 &= e_{x_1} \\ \ddot{x}_2 &= e_{x_2} \end{aligned} \tag{4.1}$$

where e_{x_1}, and e_{x_2} correspond to the acceleration process noise on in the East and North Cartesian directions, respectively, similar to that presented in [103]. Using the four state representation, $\mathbf{x} = \begin{bmatrix} x_1 & x_2 & \dot{x}_1 & \dot{x}_2 \end{bmatrix}^\top$, a linear KF is used for inference over the person model. The differential equations for the wheeled objects are:

$$\begin{aligned} \dot{x}_1 &= v\cos(\theta) \\ \dot{x}_2 &= v\sin(\theta) \\ \dot{v} &= e_v \\ \dot{\theta} &= e_\theta \end{aligned} \tag{4.2}$$

with state $\mathbf{x} = \begin{bmatrix} x_1 & x_2 & v & \theta \end{bmatrix}^\top$, where e_{x_v}, and e_{x_θ} correspond to the acceleration process noise on in the East and North Cartesian directions, respectively. An Extended Kalman Filter is used for inference over the car, bus, and cyclist models. Figure 4.4 illustrates the variables used in the coordinate system.

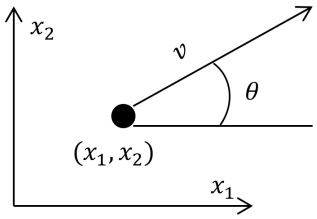

Figure 4.4: Coordinate system used in differential equations.

Using the innovation and innovation covariance, which are based on the state estimate and most recent measurement, a statistic d^2 is defined as

$$d_k^2 = \tilde{y}_k^\top S_k^{-1} \tilde{y}_k \tag{4.3}$$

where $\tilde{y}$ is the measurement innovation or residual, S is the innovation covariance, and d^2 is a chi-squared, χ^2, distributed variable with a number of degrees of freedom equal to the length of the measurement vector (2 for this scenario). The d^2 statistic, called the Normalized Innovation Squared (NIS), relates the probability of correlation between the measurement and the model as given by [104]. A bad measurement or a bad model yields a poor d^2 statistic. Engineers use this d^2 variable online to gate erroneous measurements within a selectable confidence interval, such as a 2σ, 95% bound, by comparing d^2 against the χ^2 values. To evaluate the likelihood of the model, instead of the likelihood of an individual measurement, the test statistic is averaged over k Kalman filter iterations to get $\bar{\epsilon}_k$ and compared to the associated chi-squared probability with $2 \times k$ degrees of freedom:

$$\bar{\epsilon}_k = \frac{1}{k} \sum_{i=1}^{k} d_i^2 \tag{4.4}$$

Comparing values of $\bar{\epsilon}_k$ for different models and picking the one with the highest probability yields a simple, reliable approach to infer object classification using the given measurement track. Note that while the presented formulation assumes a batch processing methodology, likelihood estimation is used in the

later developed fully joint solution to estimate object classification in real-time.

4.2.2 Joint Data Association, Tracking, and Classification

The following derives a joint perception solution for data association, tracking, and classification. The perception platform of an autonomous vehicle must take in measurements Z from a set of onboard sensors to estimate its local environment which is made up of objects O with unique dynamics which have a classification C. The discrete variable A assigns each measurement Z to the object O from which it originated. In [95] a joint solution for measurement association and object tracking is presented as a Rao-Blackwellized Particle Filter which solves $p(A_k, O_k|Z_k)$, where capital letters with k subscripts represent the set's history until time k. The sought after joint solution is that of the distribution

$$p(A_k, O_k, C_k|Z_k). \tag{4.5}$$

Some general filtering methods such as the Gaussian Mixture Probability Hypothesis Density Filter [105] exist which could estimate joint densities over different variables, but in general, no closed form solutions exist, and computational requirements for an exact filter would grow exponentially through time and would be impractical. However, there are some unique aspects of the perception problem that can be used to intelligently split the problem and make it feasible. First, the system is hybrid: object states are continuous while measurement assignments are discrete. Object classifications are discrete and typically time invariant. [1] The number of possible data assignments and time-variant

[1] A note on classification time invariance: people may exit a car, but the car has not changed classification, rather a new object has entered the observable scene. However, some classifications could be time-varying, for example the classification of an object as static (stationary) or dynamic (moving)—a parked car may start driving.

classifications grows exponentially with time, number of objects, and number of measurements, rendering exact probabilistic reasoning impossible even for simple scenarios.

Miller showed how the infeasible problem can be made feasible by using factorization and sampling techniques [95]. First, the joint solution is factored exactly as

$$p(A_k, O_k, C_k|Z_k) = p(A_k|Z_k)p(O_k|Z_k, A_k)p(C_k|Z_k, A_k, O_k). \tag{4.6}$$

This factorization provides an intuitive decoupling of the problem into discrete assignment $p(A_k|Z_k)$, object tracking $p(O_k|Z_k, A_k)$, and classification $p(C_k|Z_k, A_k, O_k)$. Decoupling helps enable solutions to tractable sub-problems, but does not fix the computational intractability due to problem exponential growth through time. By sampling the discrete data assignment density $p(A_k|Z_k)$, Monte Carlo likelihood-weighted techniques can simplify the computational complexity. Furthermore, the dynamics differential equations can be modeled in state-space giving them a Markov property that only the current state need be saved for a given object. The continuous dynamics can then be readily estimated by parametric filtering such as the Kalman Filter. By keeping only the current object state, the prior data assignment history also is not required to be kept. In [105] this same splitting was implemented via a Rao-Blackwellized Particle Filter (RBPF). For the sake of brevity, a full derivation of particle likelihood formulations including birth and death likelihoods and resampling procedures has been omitted; [105] and [95] both present thorough summaries. This paper extends these formulations by specifically adding the classification $p(C_k|Z_k, A_k, O_k)$ term.

In summary, in the RBPF formulation, each particle represents one hypoth-

esis of the scene that includes all measurements and object states through time. Measurement assignments likelihoods are drawn across all objects in a given particle. The states for each object in a particle are predicted forward in time and updated for the assigned measurement. Different particles thus represent different measurement assignment histories as well as their corresponding updates. Object births and deaths are particle specific, and particles need not have the same number of objects.

As presented in the introductory classification example, object dynamics are dependent on classification. If an object's true classification is known *a priori*, the correct dynamical model is used in the tracker. In practice, model type is not known *a priori*, so the formulation here tracks objects using a bank of filters spanning the set of possible dynamics. The probability of classification can then be written as

$$\begin{aligned} p(C|Z_k, A_k, O_k) &= \frac{p(C, Z_k|A_k, O_k)}{p(Z_k|A_k, O_k)} \\ &= \frac{p(C, z_k, Z_{k-1}|A_k, O_k)}{p(z_k, Z_{k-1}|A_k, O_k)} \\ &= \frac{p(C, z_k|Z_{k-1}, A_k, O_k)p(Z_{k-1}|A_k, O_k)}{\sum_{j=0}^{n_c} p(C = j, z_k, Z_{k-1}|A_k, O_k)} \\ &= \frac{p(z_k|C, Z_{1:k-1}, A_k, O_k)p(C|Z_{k-1}, A_k, O_k)p(Z_{k-1}|A_k, O_k)}{\sum_{j=0}^{n_c} p(z_k|C = j, Z_{k-1}, A_k, O_k)p(C = j|Z_{k-1}, A_k, O_k)p(Z_{k-1}|A_k, O_k)} \\ &= \frac{p(z_k|C, A_k, O_k)p(C|Z_{k-1}, A_{k-1}, O_{k-1})}{\sum_{j=0}^{n_c} p(z_k|C = j, A_k, O_k)p(C = j|Z_{k-1}, A_{k-1}, O_{k-1}))} \end{aligned} \tag{4.7}$$

where n_c is the number of possible classification categories. The formulation given in (4.7) is for classification that is time invariant. Time varying classification can easily be developed by extending (4.7) as the following recursion:

$$p(c_k|z_k, a_k, o_k) = \frac{p(z_k|c_{k-1}, a_k, o_k)p(c_{k-1}|z_{k-1}, a_{k-1}, o_{k-1})}{\sum_{j=0}^{1} p(z_k|c_{k-1} = j, a_k, o_k)p(c_{k-1} = j|z_{k-1}, a_{k-1}, o_{k-1}))} \tag{4.8}$$

In the case of time-varying classification, a process noise term forgetting factor could be included in the likelihood formulation in order to allow the estimate to vary based on a time-constant.

Particles within the RBPF are drawn according to a proposal density $q(A_k|Z_k)$ selected for its efficient sampling algorithms and similarity to $p(A_k|Z_k)$. Particles have a weight and diversity such that they span and represent $p(A_k|Z_k)$ as follows

$$\begin{aligned} p(A_k|Z_k) &\approx \sum_i w_k^i \delta(A - A_k^i) \\ w_k^i &= \frac{p(A_k^i|Z_k)}{q(A_k^i|Z_k)} \\ \sum_i w_k^i &= 1 \end{aligned} \tag{4.9}$$

where w_k^i is the likelihood weight of the ith particle A_k^i at time index k, and $\delta(\cdot)$ is the Kronecker delta function for discrete assignment. Given the factorization of $q(A_k|Z_k)$ as follows

$$q(A_k|Z_k) = q(a_k|Z_k, A_{k-1})q(A_{k-1}|Z_{k-1}) \tag{4.10}$$

the likelihood weight w_k^i can be expressed recursively as

$$w_k^i \propto \frac{\sum_{j=1}^{n_c} p(z_k|Z_{k-1}, A_k^i, c_{k-1}^i = j)p(a_k^i|Z_k, A_{k-1}^i)}{q(a_k^i|Z_k, A_{k-1}^i)} w_{k-1}^i \tag{4.11}$$

where the symbol $\propto$ indicates weights must be renormalized after update to maintain unity summation from equation (4.9).

Within the RBPF framework, objects are initialized with a bank of n_c KFs, one KF for each possible unique object classification. The question naturally arises of how to calculate the association likelihood for a given object in particle i against a bank of KF model classifications. Given the normalized classification probability, as is the case for a unique, mutually exclusive classification set, the

summation of probabilities of all n_c unique possible classifications for a given object is $\sum_{j=1}^{n_c} p(c_k = j|z_k, a_k, o_k) = 1$. For a parametric KF the optimal proposal density can be directly sampled. Thus the likelihoods and normalizing factor terms for sampling the overall particle filter can be given as

$$q_{\text{opt}}(a_k^i|Z_k, A_{k-1}^i) = \sum_{j=1}^{n_c} \alpha_k^i p(c_{k-1} = j|Z_{k-1}, A_{k-1}^i, O_{k-1}^i) p(z_k|a_k^i, Z_{k-1}, A_{k-1}^i, c_{k-1}^i = j) \times p(a_k^i|Z_{k-1}, A_{k-1}^i) \tag{4.12}$$

$$\alpha_k^i = \left[\frac{1}{M^i} \sum_{m=1}^{M^i} \sum_{j=1}^{n_c} p(c_{k-1} = j|Z_{k-1}, A_{k-1}^i, O_{k-1}^i) p(z_k|a_{m,k}^i, Z_{k-1}, A_{k-1}^i, c_{k-1}^i = j) \right]^{-1} \tag{4.13}$$

where n_c is the total number of classifications, j represents the selected classification, i represents the selected particle number, m represents the selected object in a particle, M^i is the total number of tracked objects in the ith particle, q_{opt} is the optimal proposal density from which particles are drawn, and α_k^i is a normalizing constant that depends on the ith particle prior to sampling.

The term $p(a_k^i|Z_{k-1}, A_{k-1}^i)$ is the transition model relating *a priori* assignment information A_{k-1}^i to the current measurement. For a generic position or velocity measurement, such as what is obtained from a radar sensor, this transition probability is uniform because previous assignments provide no information about future assignments. However, if one has a camera or point cloud, this probability could relate the previous camera's region of interest to the new measurement, or match the new lidar cluster to an existing built-up point cloud. The probability could even incorporate meta-information such as the color of the car; for example, if tracking a red car, the next camera detection could have its association likelihood modified based on how well the color matches.

The term $p(z_k|a_k^i, Z_{k-1}, A_{k-1}^i, c_{k-1}^i = j)$ is the likelihood the measurement originated for a specific tracked obstacle with the jth classification in the ith particle.

Each obstacle in the particle has a bank of n_c associated KFs, each estimating (also known as tracking) the object state according to the classification specific dynamics. For a single classification, a single KF is used and the likelihood is simply calculated from the normalized innovation, similar in concept to the normalized innovation used to classify tracks in the introductory example in Equation (4.3). With a bank of n_c KFs, the normalized innovation must be calculated for each KF in the bank. The likelihood value is then weighted by the respective classification probability $p(c_{k-1} = j|Z_{k-1}, A^i_{k-1}, O^i_{k-1})$ and summed across all n_c KF tracks in that bank. Intuitively a hard decision on an object classification is naturally made by setting the classification $C = j$ and $n_c = 1$ while not adversely affect the Bayesian formulation.

A high-level description of the joint data assignment-tracking-classification algorithm is given as follows.

1. Draw an initial set of particles $A^i_0 \; \forall \; i \; \exists [1, N]$.
2. Predict all obstacles in each particle forward in time to the next measurement z_k to yield a parametric representation of $p(O_k|Z_{k-1}, A^i_{k-1})$.
3. Randomly sample a new set of data assignments for z_k from the optimal proposal density according to Equation (4.12) and the given sampling procedure.
4. For each object in each particle, update the bank of parametric-tracking filters to yield $p(O_k|Z_k, A^i_k) \; \forall \; C$ and update the classification $p(c_k|Z_k, A_k, O_k)$.
5. Update particle weights according to $w^i_k \propto w^i_{k-1}/\alpha^i_k$ and (4.13), and renormalize the weights to sum to unity.
6. Resample particles to keep the filter well conditioned, if necessary.

Finer grained classification requires sensor-specific data processing and techniques. For example one could build a 3d colorized point cloud of the object and compare it to a library. The formulation included here natively supports sensor-specific output of object classification along with weak classifiers typically used in a boosting framework [102]. Boosted weak classifiers and sensor-specific classification information could be incorporated into the likelihood formulation of $p(z_k|c_{k-1}, a_k, o_k)$. Experimental studies in the results section of this paper demonstrate classification performance informed solely with lidar clustering and vision processing of object classification.

4.2.3 Classification with Multiple Hypothesis

Vision-based car detections provide vehicle heading in addition to locating vehicle's bounding box within the scene [106]. Typically the detector can correctly extract the major-axis line of the vehicle, but it often confuses the front and back of the car along that line. This causes the angular heading measurement to have a bi-modal distribution with a main peak along the forward direction of the car and a secondary smaller peak in the 180 degree reverse direction. Handling this distribution as a single Gaussian would require a very large distribution over the angle range of $-\pi$ to π radians. Furthermore, the propagation of the position dynamics is coupled to both the heading and ground speed, so a large covariance in the heading estimate causes the filter's overall position and velocity estimates to become uncertain and degrades future measurement association performance.

Multiple options exist for handling multi-modal distributions, such as Gaus-

sian Sum Filters [107] or Particle Filters [108]. An alternative approach is to add a state variable for vehicle direction, either forward or reverse, and treat the measurement in two parts 1) as the angle to a line and 2) a binary variable corresponding to the forward direction along the heading line. This measurement splitting allows for a simple formulation with minimal additional computation by augmenting the Kalman Filter continuous-state tracker with a Bayesian classifier for the discrete random variable of the vehicle's forward direction. The following formulation allows for tracking vehicles driving forward or reverse. It correctly distinguishes the true vehicle orientation with the only requirement being that the camera (or other sensor) has a weak classifier of vehicle forward direction; that is, the classifier must correctly distinguish forward from reverse in more than 50% of its heading detections.

Along the central axis of the car, two classifications H for heading exist as the set $h \exists \{0 = \texttt{reversed}, 1 = \texttt{correct}\}$. The classifier $p(H|Z_k)$ is written as

$$\begin{aligned} p(H|Z_k) &= \frac{p(H, Z_k)}{p(Z_k)} \\ &= \frac{p(H, z_k, Z_{k-1})}{p(z_k, Z_{k-1})} \\ &= \frac{p(H, z_k|Z_{k-1})p(Z_{k-1})}{\sum_{h=0}^{1} p(H = h, z_k, Z_{k-1})} \\ &= \frac{p(z_k|H, Z_{k-1})p(H|Z_{k-1})p(Z_{k-1})}{\sum_{h=0}^{1} p(z_k|H = h, Z_{k-1})p(H = h|Z_{k-1})p(Z_{k-1})} \\ &= \frac{p(z_k|H)p(H|Z_{k-1})}{\sum_{h=0}^{1} p(z_k|H = h)p(H = h|Z_{k-1})} \end{aligned} \tag{4.14}$$

The last line can be read as the k^{th} detection likelihood times the prior divided by a normalizing factor. As long as the camera's classification detection probability for a given detection exceeds 50%, this sequential estimator correctly classifies the probability of the heading. Additionally, if initialization is based on object

location with respect to lanes, the initial prior $p(C|Z_0)$ could be nonuniform and heavily weighted for a car obeying rules of the road and traveling in the direction of traffic flow. Given a uniform prior, the maximum *a posteriori* of the classifier simplifies to counting the number of heading measurements aligned with forward direction h_1 and reverse direction h_0, and classifying the vehicle direction with the largest number of counts.

A description of the implementation follows. For vehicle objects initialized by a camera detection, the filter is unmodified: the object's relative heading is initialized to match the measurement z_θ. However, for initialized objects, camera detection measurement updates of vehicles are modified; the measurement residual is computed for both the actual heading measurement z_θ and the reversed heading measurement $z_\theta + \pi$. The smaller residual measurement is applied and the respective count for heading aligned measurement h_1 or heading reversed measurement h_0 is incremented. After each heading measurement is applied to an object's KF, a classifier is run by simply selecting the maximum heading count classification $max(h_1, h_0)$. If the classifier finds the vehicle forward direction is reversed, then: the sign of the speed estimate is reversed, the heading angle is reversed 180 degrees, and the KF state covariance is updated accordingly.

4.3 Experimental Hardware, Sensors, and Sensor Processing

Experiments in this paper utilize Cornell's DUC entry vehicle, a 2007 Chevy Tahoe dubbed Skynet [109]. The vehicle contains an assortment of radar, lidar, camera, GPS, odometry, and inertial measurement sensors. Internally, the vehi-

cle features a 19U server rack for experimental algorithm deployment and data storage; experiment data can be collected for offline playback and evaluation. Typical operation with all sensors generates 350GB of raw uncompressed data per hour. Hardware upgrades since the DUC [109] include a Point Grey Ladybug3 360 degree field of view spherical camera, external waterproofing of all sensor mounts and wiring, and upgrading of onboard computing and storage rack-mount servers.

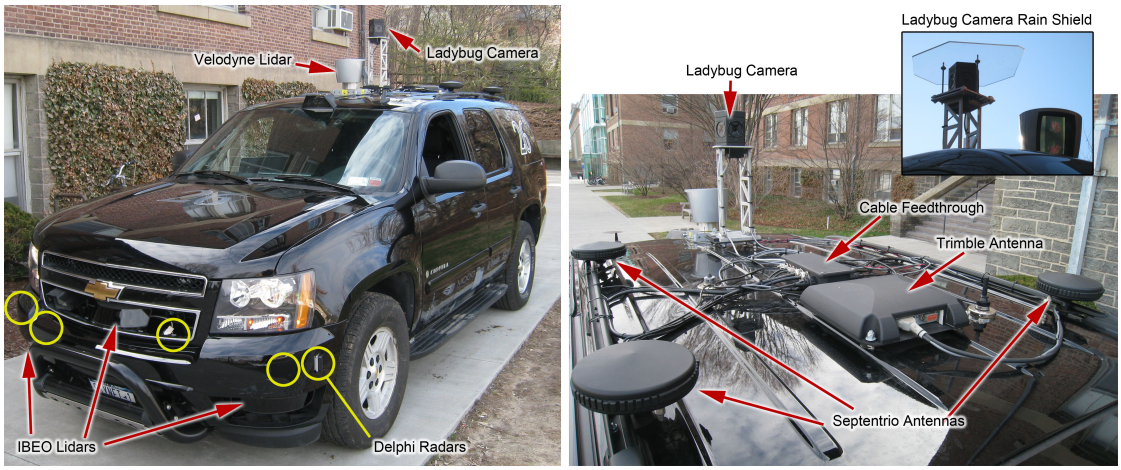

Figure 4.5: Skynet sensor suite.

Figure 4.5 shows the vehicle exterior with sensors well integrated with minimal external protrusions and wiring. Quality installation is critical to reliable performance in rain and snow. In addition to local environment perception sensors, Skynet is equipped with an attitude and position estimation system composed of a Litton LN-200 IMU, ABS wheel encoders, Septentrio PolaRx2e@ GPS receiver with three roof mounted antennas in an "L" configuration, and a roof mounted Trimble Ag252 GPS receiver. The LN-200 is floor mounted on the vehicle centerline above the rear axle. The Septentrio provides 5Hz synchronized GPS measurements of raw pseudorange, Doppler shift, and carrier

phase to satellites and decodes WAAS signal. The Trimble reciever decodes high-precision (HP) OmniSTAR differential corrections at 10Hz with 10cm accuracy. A pose estimator described in [110] combines sensor measurements to utilize strengths of each sensor and diversity to generate a robust attitude and position estimation solution.

A 360 degree field of view Velodyne HDL-64E S1 lidar unit with 64 vertical laser scans is mounted on Skynet's roof. Velodyne lidar returns were used to detect the ground plane. Three IBEO XT Lidar units with 4 laser scans each are mounted on the front bumper. Ibeo lidar points are actively classified by Ibeo's proprietary software as object, rain, ground, or dirt, where dirt refers to lens cover fouling.

A Ladybug3 LD3-20S4C-33B spherical color camera is mounted behind and above the Velodyne for a clear view of surroundings. Narrow vertical struts provided rigid mounting and minimal blockage for the Velodyne lidar. The camera has five lenses pointed horizontally outward in a pentagon configuration and one lens pointed vertically. The camera lenses are factory calibrated to export 360 degree spherical or cylindrical projections with vendor provided software. In lieu of building a complex mechanical wiper device, for experiments in this paper, a clear plexiglass cap shown in Figure 4.5 is mounted above the Ladybug camera to minimize lens fouling during precipitation. Eight Delphi forward-looking millimeter-wave radar units are mounted around the vehicle; each includes proprietary black-box onboard processing to generate tracks in the form of object bearing, range, and radial speed. Tangential speed cannot be measured by radar and is not estimated by the radar unit's onboard processing. All sensor data is timestamped with 100usec accuracy via ethernet connected

micro-controllers and a pulse clock synchronized to the LN-200 IMU. Figure 4.6 shows the sensor coverage; overalapping regions of Ibeo lidar and Ladybug camera in front of Skynet enable comparison studies between sensor modalities. The effect of processing resolution for Ladybug imagery is studied in the results section. The remaining results sections use Ladybug imagery downsampled to one quarter resolution for near real-time detection processing due to computational constraints. Full resolution processing enable car detections up to 70m range and 40m range person detections; downsampling reduces the active range to 15m for cars and 10m for people. A 20m semicircle was used for quantitative evaluation comparisons.

Skynet is outfit with an occupancy grid as a safety catch-all to prevent running into objects that did not get tracked in the RBPF. This occupancy grid is a common commercial feature often implemented with sonar or radar. On Skynet this is implemented with lidar.

The classification formulations given in equations (4.7) and (4.14) can be numerically sensitive to machine precision underflow. Tracking errors induced from temporary filter instability, poor initial conditions, or a series of poor measurements can quickly drive a model's probability, for example $p(C = j|Z_k, A_k, O_k)$ or $p(H = h|Z_k)$, to zero in machine-precision—that is, the incorrect model could have 100% classification probability and the sequential estimator numerically multiplies future correct classification measurements by zero. The tracking filter may improve its estimate of object state, but the recursive classification multiplication is stuck at zero or one, effectively making an ad hoc classification due to machine precision underflow. Implementation solutions include performing the probability calculation in log-likelihood units or thresh-

olding the minimum or maximum classification probability values in the range $0 + \epsilon < p(\cdot) < 1 - \epsilon$ for some small value of ϵ, which was done in Skynet.

All perception algorithms aside from the camera detector ran in real-time on a variety of Intel two and four core x86 64-bit processors in a Windows 7 environment. For simplicity, primary function algorithms are run on individual computers and data is shared between sensors and algorithms via UDP Multicast across a Hewlett-Packard V1910-48G managed gigabit ethernet switch. The RBPF joint data association, tracking and classification routine is the most computationally intensive algorithm and reliably runs in real-time on an i7-3820 with 8 particles; that is eight complete hypothesis of the entire scene. Lidar clustering is run on an i7-930 while the remaining routines are run on an assortment of i5 and i7 processors. The following sections detail raw sensor processing, developed since the DUC [109] and [95], to detect vehicles and people with the Ladybug camera and to process lidar returns for person-sized and car-sized clusters.

4.3.1 Vision-based Detection

The field of computer vision has been rapidly advancing. Recent studies have shown improved detection and classification rates for cars [88] and pedestrians [111], [112]. The vision-based detection system presented below is a state-of-the-art detector for cars and pedestrians which also detects the heading of vehicles. Vehicle heading detection, a unique feature of the detector, is computed from each still frame from the camera, and thus does not require multiple frames or tracking through time. The following section explains how the Deformable

Parts Model (DPM) technique [84] trained on existing datasets was used for both car and pedestrian detection.

The vision-based detection subsystem makes use of a Point Grey Ladybug 3 spherical camera mounted on top of the vehicle. Images are acquired at 6.5 frames per second synchronized with the vehicle's global clock and stitched into spherical panoramas using the vendor-provided software. Each spherical panorama is then reprojected into 8 separate rectilinear virtual cameras at 45 degree increments; reprojections are referred to as tiles.

Each tile is passed through two state-of-the-art detections, the first being a car detector and the second being a person detector. The car detector was first introduced in [106]. Both detectors make use of the Deformable Part Model (DPM), a technique for robustly detecting and localizing objects under varying viewpoint and illumination conditions by analyzing distributions of image gradients [84]. Local gradient statistics are aggregated to form rigid *parts*, small square patches that often have some semantic meaning, such as a wheel on a car. The model encodes the rest position of each part with respect to a root coordinate system, but additionally encodes an energy required to deform each part. For example, a single DPM can encode a large variety of car makes and models despite variations in shape or size by modeling how one part varies from a rest position according to the training set.

In addition to detecting and localizing passenger vehicles relative to the ego-vehicle, the car detector also predicts an orientation. A total of 16 separate DPM models are used to cover a set of orientations. Training examples are derived from three separate datasets: VOC2007 [113] – an Internet dataset with 2D bounding box annotations, KITTI [114] – an autonomous vehicle dataset with

3D bounding box and orientation annotations fitted to LIDAR point clouds, and NYC3DCars [106] – an Internet dataset with 3D bounding box and orientation annotations built by estimating scene geometry and asking annotators to place 3D models in the reconstructed scene. The detector from [115] is used for person detections.

After each tile has been processed, detections are aggregated per panorama and car orientation estimates are transformed from tile-space coordinates to vehicle-space coordinates. Non-maxima suppression is applied to threshold car and person detections. Detection boxes of pedestrians and cars are calculated in spherical angular coordinates. Object bearing relative to the ego vehicle is calculated from the centroid of the detection boxes. For autonomous vehicle tracking purposes, an estimate of object range is also helpful. By assuming the ego vehicle is oriented parallel to and contacting with the ground, and that all detected objects are in contact with the ground, a flat ground plane model and trigonometry can be used to estimate the nearest range in meters to the object and the object's width in meters.

The described algorithm is computationally intensive. An 8 core E5-2660 Xeon CPU and Nvidia GTX980 GPU were utilized to accelerate color processing, image resizing, and JPEG exporting of frames from recorded video sequences. Code implementation of the detection processing was performed on Amazon EC2 Cloud computing platform using 200 of their c3.large machine instances which contain 2 virtual CPU's (vCPU) with 3.75GiB of memory. The DPM is not optimized for computational efficiency or speed; thus, unoptimized implementation was run on a server farm. Full resolution panoramas from the Ladybug camera software were exported at 8000x4000 pixels, and tiles were exported

at 2048x2048 resolution. Processing full resolution tiles through the DPM consumes all 4GB of RAM and takes 5.5 minutes per tile on a single machine. Reducing panoramas to 2000x1000 pixels and tiles to 512x512 drops computation time closer to real-time at 7 seconds per tile. A thorough analysis of imagery resolution and detection performance including recall and precision rates is included in the results section.

4.3.2 Lidar Segmentation

Car-sized clustering was unchanged from the DUC as described in [109]. Raw lidar points, classified as objects by Ibeo's proprietary software, are further trimmed by removing any point within 0.3m of the detected ground plane. The remaining points are clustered into groups which have minimum size and maximum horizontal point spacing of 0.5m threshold and then a second time at a 1.0m threshold. Only objects that pass both clustering thresholds and whose maximum dimension is less than 15m (the maximum typical length of a bus) are considered 'stable' clusters and passed to the joint perception algorithm. Clusters must contain at least seven points and must have at least one point projecting over 1.0m in height. Resulting clusters are classified as 'car-sized'.

New person clustering is implemented by extending Laplacian of Gaussians (LoG) filtering typically used in image processing to lidar processing. Laplacian filters are derivative filters applied to images to find edges. Gaussian filters blur or smooth an image. For edge detection in traditional image processing, a normalized 2D LoG filter is built such that it calculates a large magnitude before and after an edge; there is a sign-change at the edge. With lidar, most

objects appear as edges, but walking or standing people typically appear as a dense cluster of points separated from surrounding returns. In order to detect people, the filter is modified to find a person-sized group of points which are isolated from neighboring lidar returns. By constructing a normalized 1D LoG and sweeping the filter radially around a point, a 2d filter is constructed that has a negative mean value. The filter response is a negative value to a uniform field, a negative value to an edge, and a large positive value to a person-sized cluster isolated from its surroundings.

Object point returns are projected into a horizontal plane and grouped in a square grid with 25cm sized cells. The heuristically modified LoG response is computed per lidar return

$$L(r) = \frac{1}{\pi\sigma^2}\left[1 - \frac{r^2}{2\sigma^2}\right]e^{\frac{-r^2}{2\sigma^2}} + \frac{0.15}{\pi\sigma^2}\left[1 - \frac{r^4}{2\sigma^2}\right]e^{\frac{-r^2}{2\sigma^2}} \tag{4.15}$$

where r is the radius from filter center in the horizontal plane and $\sigma = 0.45$m. A plot of the filter magnitude versus radius is shown in Figure 4.7. By computing the filter response at each grid cell as $\sum L(r)$ for all lidar returns, a person cluster can be identified. Figure 4.8 shows an example uncluttered roadway scene containing 1 person after being processed by the LoG filter for a person. The positive peak is the person's location; the negative peak is the area neighboring the person; and other negative areas correspond to other objects and edges from the scene. Non-maximum suppression and a minimum positive signed amplitude threshold is applied across the grid to select peak locations for person-sized clusters. Both person-sized and car-sized lidar clusters are then passed to the joint association, tracking, classification algorithm. Recall and precision rates are included in the results section.

4.4 Simulation and Experimental Results

The results section of this paper first motivates probabilistically adding classification to the tracking problem. Kalman Filter inference of object classification is demonstrated in both Monte Carlo simulation and experimental data by running a bank of filters on object measured data. The experimental portion uses data collected with a low accuracy hand-held GPS unit from pedestrian, biker, car, and bus objects traversing around Cornell's campus.

Next, the joint data association, tracking, and classification algorithm performance is evaluated using Skynet—Cornell's AGV entry from the DUC. Repeatable scenarios of intersection encounters were conducted in multiple weather conditions and recorded with camera 'C', lidar 'L', and radar 'R' sensors. Quantitative evaluations were conducted in post-analysis with the reduced sensor sets L+R and C+R and complete sensor set C+L+R in order to evaluate if cameras can replace lidars for AGV applications. Performance of vision heading detection inclusion in the estimator and selection of the number of particles used in the RBPF are analyzed. Ground truth of two pedestrians and one vehicle enables quantitative evaluation of tracking and classification performance. The joint solution demonstrates robustness to all weather conditions.

Performance of reduced sensor set evaluations from the quantitative experiments provides insight into individual sensor performance in different weather conditions in staged scenarios. Additional non-staged qualitative experiments were conducted by driving Skynet through traffic around downtown Ithaca, NY to more broadly evaluate sensor and perception algorithm performance in varied weather conditions.

4.4.1 Monte Carlo Simulations: Joint Tracking-Classification

Monte Carlo simulations were conducted to evaluate classification performance of two categories, person and cyclist, given truth data generated with a dynamics function that exactly matches the dynamics modeled in the KF.

Synthetic tracks, 50 seconds in length, sampled at 1Hz, of both people and cyclists were generated from the KF modeled dynamics functions. Motion occurred from randomly selected initial conditions and process noise; synthetic measurements were created by adding Gaussian measurement noise to the synthetic tracks. Measurement noise was randomly drawn per track inside the MC truth simulation; each track is considered an MC iteration. Measurements from each of these tracks were then run through both KFs and classified based on the highest χ^2 probability. Both filters are initialized by setting the initial position equal to the first position measurement and the initial velocity vector tangent to the line connecting the first and second position measurement. The limit of the 50^{th} percentile χ^2 cumulative density function (cdf) should approach the number of degrees of freedom in the measurement vector which is two. By averaging the χ^2 values over multiple KF updates, a distribution with $n_{\mathtt{DOF}}$ degrees of freedom is generated as follows

$$n_{\mathtt{DOF}} = kn_z \tag{4.16}$$

where k is the number of measurements and n_z is length of the measurement. For the MC simulations, $k = 50$ measurements of dimension $n_z = 2$ provides a distribution with $n_{\mathtt{DOF}} = 100$ degrees of freedom. The 50^{th} percentile χ^2 cdf can

be calculated as follows

$$\begin{aligned}\chi^2_{\texttt{cdf}} &= \chi^2_{\texttt{cdf}}(\texttt{percentile}, n_{\texttt{DOF}})/n_z \\ &= \chi^2_{\texttt{cdf}}(0.5, 100)/50 \\ &= 99.3/50 = 1.986\end{aligned} \tag{4.17}$$

This 50^{th} percentile is shown as a black dashed line in Figure 4.9 along with the χ^2 averages for each test run. The green trace is the likelihood the track was generated by a person while the blue trace is the likelihood the track was generated by a biker. The top plot track iterations were generated with cyclist wheeled dynamics while the lower plot track iterations were generated with the person walking dynamics.

Quantitative numbers summarizing the plot are show in Table 4.1. The χ^2 average listed in the table is the average value of the traces from Figure 4.9 over all 100 iterations. The true model classification averages close to the ideal 2.0 mean while the incorrect model is farther from 2.0; these results show that, given enough data and the models, the biker and person can be inferred. The simulation correctly classified 100% of cyclist and 79% of the person tracks.

Table 4.1: Monte Carlo simulation results.

Track Generation:	Average χ^2		χ^2 Classification Percentage	
	Cyclist	Pedestrian	Cyclist	Pedestrian
Cyclist	2.0897	7.9631	100	0
Pedestrian	1.5916	1.8784	21	79

4.4.2 Experimental Results: Joint Tracking-Classification

Experiments studying joint tracking and classification were conducted by building KF models of four classifications: pedestrian, biker, car, and bus. A low accuracy handheld GPS with 1.1 meter standard deviation of error recorded position data from each object. A portion of the data was used to estimate the process noise parameters for each class. Classification performance was analyzed on the remaining data to demonstrate that accurate classification can be accomplished using only GPS position measurements of the objects.

All data was collected in Ithaca, NY around Cornell University campus and the Ithaca Commons. Euler integration is used to predict the model of the continuous time obstacle dynamics at 1Hz intervals. GPS data was collected using a Locosys GT-31 handheld unit at 1Hz frequency and recorded in NMEA-GGA sentence format, which has approximately 18cm of quantization error in Ithaca, NY. Measurement noise for the sensor was computed using data collected from a stationary sensor to establish a covariance matrix R in meters squared as follows

$$R = \begin{bmatrix} 1.2 & 0.1 \\ 0.1 & 1.2 \end{bmatrix}. \tag{4.18}$$

Car data was collected using a 4-door sedan while bus data was collected while riding on a local commuter bus. The filters are not expected to distinguish non-moving targets. Because some of the driving data contains time waiting at traffic lights, any data sections where the GPS unit moved less than 25cm on average between downlinks was removed from the evaluation set. A holdout set of the data 5 minutes in length was used with a Matlab minimization routine to estimate the process noise covariance values for the four models and is given in Table 4.2.

Table 4.2: Process noise values.

Model:	Pedestrian	Car	Bus	Cyclist
Acceleration $\mathtt{m/s^2}$	0.04	0.6	0.4	0.31
Rotation Rate $\mathtt{deg/s}$	N/A	15	15	15

Table 4.3: Successful classification rate from experimental GPS data.

Model:	Pedestrian	Car	Bus	Cyclist
Percentage	100%	99%	99%	50%

From each experimentally recorded data set, 100 sections of 50 consecutive GPS data points were randomly selected and fed into the bank of Kalman filters enumerating each classification. Classification was inferred using all four KFs and reported in Table 4.3. Pedestrian, car, and bus were correctly classified for at least 99% of the tested tracks by the χ^2 statistics. The cyclist classification rate was much lower, in part because the collected cyclist data contained more stops and starts; of the three other classes, the cyclist was most often mis-classified as a pedestrian. In summary, the example with experimental data shows reliable classification performance solely from position data, highlighting an example benefit possible with joint tracking and classification solutions.

4.4.3 Experimental Data Collection

Several datasets were collected in order to study key elements of the solution, including algorithm parameters, sensor types, and weather. In the first dataset, experiments were conducted with specified encounters on a closed course of one vehicle and two pedestrians and were repeated multiple times in different weather and lighting conditions to understand how weather affected sensor and algorithm performance.

Quantitative analysis of the controlled experiments is enabled by using a 1999 Chevrolet Suburban as a ground truth vehicle. The Suburban is outfit with the same GPS and inertial measurement sensors as Skynet, which resolve centimeter-accuracy position estimation reported at 100Hz of the vehicle obstacle. By differencing high precision pose estimates of the Suburban from Skynet, a relative truth dataset was obtained for the various encounters. Real-time sub-meter GPS sensing for pedestrians was not available. Instead a combination of a pre-surveyed path, low precision GPS, and a camera tracker were used. The pre-surveyed path was marked for pedestrians in the closed course. The time-synchronized camera recorded accurate timing of the pedestrian's location on the surveyed path, enabling generation of sub-meter accuracy truth data for the pedestrians in the closed course. All truth data is synchronized in post-processing using GPS recorded timing information.

Three experimental scenarios in Figure 4.10 quantitatively evaluate the repeatability and performance of the RBPF across four different weather scenarios. For Scenario A, the Suburban and Skynet drive towards each other and stop at an intersection; Skynet turns as the Suburban drives straight through the intersection. For Scenario B, Skynet and the Suburban approach and stop

at an intersection, and the Suburban crosses in front of Skynet. For Scenario C, Skynet approaches and stops at an intersection, and two pedestrians cross in opposite directions in front of Skynet; Skynet proceeds through the intersection after pedestrians cross. For each scenario five experimental trials were conducted for each weather condition. The experiment was conducted over four different weather condition categories: Sunny, Night, Wet & Cloudy, and Snow & Rain. Visual data and example detections for all four weather condition categories are shown in Figure 4.11. The Wet & Cloudy trials were recorded after a rain storm; the ground was wet but no precipitation was present and the sun was occluded by clouds. This condition most closely resembled the conditions on the second day of the KAIST competition that resulted in two autonomous vehicles crashing [75].

By logging all timestamped sensor recordings, data could be replayed through the algorithms for studies such as a reduced set of sensors or modifications to the RBPF. Quantitative evaluation is achieved by using the most likely particle from the RBPF and comparing the closest tracked object to the truth object, which could be a pedestrian or Suburban depending on the scenario. The closest object to truth is considered correctly tracked if the truth's and estimate's object centroids are within 2m. The RBPF output is recorded at 10Hz for evaluation in a 20m semi-circle in front of Skynet. The results tables in this section contain columns with the following specific definitions: Object Tracked is the fraction of time an estimated object from the RBPF overlapped the truth location; Range and Bearing Errors are reported as root-mean-square (rms) statistics of the range to closest point and mean bearing respectively; Correct Classification is the fraction of time that a tracked object was correctly classified, while Mis-Classification is the fraction of time a pedestrian was errantly classified as

a car or vice versa; and Number of Returns is the number of time instances that the estimated object overlapped the truth object. Unclassified time fraction is not reported in the tables but may be easily calculated; the sum of Correct, Mis-, and Un- classified must equal one. An example encounter of an object being properly tracked in range and bearing is shown in Figure 4.12.

In addition to the controlled dataset, an additional experiment in an uncontrolled environment in Sunny conditions was conducted by driving Skynet through Cornell's campus from B-Lot past the Engineering Quad through CollegeTown for 7 min 13.4 sec. This experiment contained a high density of pedestrian and vehicle traffic and is used in the following results sections to compare vision resolution performance, sensor precision and recall rates, and particle count selection for the RBPF filter. Truth data was unavailable for this uncontrolled experiment.

4.4.4 Particle Count Selection for Joint Data Association, Tracking, and Classification

A key design consideration for the RBPF is selecting an appropriate number of particles. Each particle represents a full hypothesis of the measurement associations and object states in the local environment. In Miller's original RBPF [95], multiple hypotheses helped model ambiguity associated with data association. For Cornell's DUC entry, a total of four particles ran in real-time and adequately captured variability in data association, primarily due to the wide spacing between vehicles. With the addition of classification, the RBPF can help model ambiguity regarding object type classification, such as car or pedestrian,

Table 4.4: RBPF performance for different numbers of particles. *For all table results:* Object Tracked is the fraction of time an estimated object from the RBPF overlapped the truth location; Range and Bearing Errors are reported as root-mean-square (rms) statistics of the range to closest point and mean bearing respectively; Correct Classification is the fraction of time that a tracked object was correctly classified, while Mis-Classification is the fraction of time a pedestrian was errantly classified as a car or vice versa; and Number of Returns is the number of time instances that the estimated object overlapped the truth object. Unclassified time fraction is not reported in the tables but may be easily calculated; the sum of Correct, Mis-, and Un- classified must equal one.

Number of Particles	Object Tracked	Range Error (m)	Bearing Error (rad)	Correct Classification	Mis-Classification	Number of Returns
1	0.993	1.698	0.068	0.917	0.001	1484
4	0.998	1.545	0.076	0.870	0.000	1496
8	0.998	1.544	0.077	0.875	0.000	1455

or capture other binary object characteristics such as a vehicle's true heading direction.

For more complex scenes, such as those considered here with both cars and pedestrians, the number of particles may need to be higher. The first study of particle counts uses the sunny dataset; all trials from all scenarios were used, and the RBPF was run with 1, 4, and 8 particles to study the effect on performance. Results in Table 4.4 show negligible performance improvement comparing 1, 4, and 8 particles. The association ambiguity was negligible for the largely spaced objects in these intersection scenarios; hence performance was similar across the different number of particles, and controlled intersection experiments could be analyzed with only a single particle.

The second particle study used the data recorded driving through Cornell's campus and CollegeTown, with a more complex scene with numerous pedestrians and vehicles making data association and classification less obvious. Truth data was unavailable for this uncontrolled experiment. Data is played back

with two configurations of algorithms: 1) Miller's original Clustering and RBPF algorithms from the DUC, and 2) the extended Clustering and RBPF with classification algorithms presented in this paper. Comparisons between the two configurations help understand how selection of RBPF particle count is affected by the addition of classification. Given the RBPF is extended to classify two object types, namely cars or pedestrians, it was initially hypothesized that doubling the number of particles used in the original DUC RBPF from four to eight might provide adequate performance. Data is passed to both algorithm configurations and run with 1, 4, 8, 12, 16, and 20 particles. A playback using 50 particles for each configuration is treated as the benchmark for comparison. Errors in object counts are computed by differencing individual runs against the respective configuration's 50 particle count run.

Table 4.5: Joint Data Association and Tracking (DUC Tracker)

	Joint Association & Tracking	**Joint Association, Tracking, & Classification**	
Number of Particles	**RMS Error: Number of Objects**	**RMS Error: Number of Objects**	**RMS Error: Num. Classified Objects**
1	2.76	4.39	1.94
4	2.45	4.10	1.80
8	2.32	3.80	1.80
12	2.21	3.57	1.84
16	2.33	3.54	1.64
20	2.31	3.18	1.78

Table 4.5 and Figure 4.14 show the results from Miller's DUC tracker (configuration 1) which used only car-sized lidar clusters, no Ladybug imagery, and no classification; and the joint classification-tracker developed here (configuration 2) including clustering, imagery, and joint data association, tracking, and classification routines. The truth simulation run tracked an average number of 34 objects for configuration 1 and 66 objects for configuration 2, of which on average 8 were classified as cars and 4 were classified as pedestrians. Plots of the cumulative distribution function (CDF) of errors in the number of tracked objects is presented in Figure 4.13.

For configuration 1, joint association and tracking, in accordance with [95], increasing particle count from 1 to 4 improves performance; performance plateaus at 8 particles. For configuration 2, adding classification along with vehicle heading ambiguity, pedestrian lidar clusters, and increased number of camera returns, provides significant additional complexity. The RBPF filter continues to improve performance with increased number of particles out to 20 particles, as highlighted in Figure 4.14. Put another way, errors in overall (classified + non-classified) object counts decrease with more particles. However, errors in the number of classified objects plateaus at 4 particles, so the addition of classification required a minimum of 4 particles. Objects that have been classified are well-tracked; higher particle counts provided negligible benefit for these well-tracked objects. Four particles is adequate for classified objects because it is rare for an object to simultaneously have ambiguous data association, classification, and heading direction. The benefit in increasing to a higher number of particles was observed for objects that do not get classified such as shubbery, buildings, or distant pedestrians/vehicles that have sparse measurement returns.

Configuration 2 tracks a larger overall number of objects than configuration 1 which contributes to configuration 2's higher overall error rate of number of tracked objects. However, the error in number of classified objects tracked in configuration 2 is less than the error in number of unclassified objects in either configuration; thus, classification helps to improve the overall perception system's data association and tracking performance.

The number of reasonable and therefore possible associations and object classifications can increase due to closely spaced objects like pedestrians, which makes selection of an optimal number of particles non-obvious. When considering more than 4 particles, a trade-off is reached between increased computational complexity and probabilistic fidelity. In general, the results of classified object track count errors plateauing at 4 particles implies that 4 particles is adequate for classified object tracking in urban driving scenarios, but additional particles support more congested scenes. For the other studies in this paper, 8 particles was chosen due to its balance between performance and real-time capability in the author's C++ implementation. All remaining experiments, both controlled intersection scenarios and urban downtown driving, are run with 8 particles.

4.4.5 Controlled Experiments: Sensor Sets in Joint Data Association, Tracking, and Classification

Examination of sensor segmentation performance, along with the output's integration into the joint association, tracking, and classification algorithm is presented in this section. The first study is the performance of the vision detection

Table 4.6: Camera processing per resolution. For reference, 2k is similar in resolution to 1080p HDTV.

	1k	**2k**	**4k**	**8k**
Actual Resolution:	1024x512	2048x1024	4096x2048	8000x4000
Tile Resolution:	256x256	512x512	1024x1024	2048x2048
Car Detection Range:	7m	15m	32m	70m
Person Detection Range:	5m	10m	20m	40m

Table 4.7: Sensor recall and precision rates. Recall is computed over sensor's range.

Sensor, Condition	**Car Recall**	**Car Precision**	**Person Recall**	**Person Precision**
Lidar, Sunny	91%	51%	88%	55%
Camera, Sunny	85%	99%	70%	90%
Camera, Rainy		93%		76%

algorithm as a function of image resolution. As described in the computer vision section, camera images were downsampled due to computational restrictions. Table 4.6 presents four levels of resolution, along with the observed car and pedestrian detection ranges. Full resolution 8k video is reduced to the near real-time version of 2k which contracted the car detection range from 70m to 15m and person detection range from 40m to 10m. A comparison of full and reduced resolution detections from the Ladybug camera is shown in Figure 4.15 for two scenes. In the upper photo pair, 2 people are detected in the downsampled image versus 13 in the full resolution image. In the lower photo pair, 4 cars are detected in the downsampled image versus 26 in the full resolution image. Downsampling images reduces the number of pixels describing an object; given enough downsampling any object will become obscured. In the shown images, smaller and more distant objects are not detected in the downsampled versions. In practice, reducing camera resolution reduces object detection range.

Table 4.8: Sensor detection rates averaged per sensor return.

Sensors Used	Objects Detected	Cars Detected	Persons Detected
C-1k	0.61	0.46	0.15
C-2k	1.98	1.53	0.45
C-4k	10.50	4.52	5.98
C-8k	35.44	8.12	27.31
L	12.04	9.26	2.78

Table 4.9: Sensor object tracking and classification average rates.

Sensors Used	Objects Tracked	Cars Tracked	Persons Tracked
C-1k + R	41.10	3.00	0.04
C-2k + R	42.86	7.31	0.22
C-4k + R	50.02	14.14	2.49
C-8k + R	63.16	27.99	4.35
L + R	63.11	0.27	3.20
C-8k + L + R	81.86	27.67	9.44

Using a holdout set of data, recorded from driving 3.0 miles from Cornell's campus through College Town to downtown Ithaca, NY, true classification rates of precision were computed for car and person detection routines by hand labeling at least 300 detections. Summary statistics of recall and precision are presented in Table 4.7. When normalized by sensor range, recall and precision rates were found to be independent of camera resolution. During clear daytime conditions, detections had a correct classification rate of 99% for a car and 90% for a person. During rainy conditions, true classification rates rates dropped to 93% for car and 76% for pedestrian. Thus, one car detection or two person

detections are sufficient for a 95% confident classification of a respective car or person object. Clearly, as computation becomes faster over time, the performance of the visual detector—particularly in range—will improve dramatically. Recent developments such as [89], [111], and [112] have demonstrated real-time implementation by intelligently selecting regions of interest for processing and by utilizing GPU acceleration for DPM implementation.

True detection rates (precision) for person and car classified lidar clusters are calculated by analyzing performance of clustering across clear day and night weather scenarios using a holdout dataset collected from Cornell campus, College Town, and downtown Ithaca, NY. Lidar clustering maximum range is approximately 70m. The lidar clustering routines are designed based on object size, not object feature extraction; thus similarly sized non-person and non-car objects are clustered as person-sized or car-sized respectively. In vehicle experiments, the car clustering routine gave a true classification precision rate of 79% for car-sized objects. Occasional errors occurred from clustering shrubbery or buildings as large vehicles. In total, 36% of objects correctly identified as car-sized were not in fact cars; the true classification rate for cars was 51%. Person clustering routine gave a true classification rate for person-sized objects of 89%. Signs, fence posts, and telephone poles were the most common sources of errant person clusters because their lidar signature is similar to that of a human; false positives accounted for 36% of the person-sized detections. The true classification rate of actual persons was 55%. For the purpose of demonstrating joint perception solution in varying weather conditions, a simple robust clustering method was selected over high-fidelity, computationally complex, and brittle methods. Achieving 95% confidence of object classification using the presented clustering routine's true classification rates of 51% for cars and 55% for pedes-

trians with the Ibeo Lidar units reporting at 12.5 Hz requires tracking uniformly classified clusters for 6 seconds to classify a car or 1.2 seconds to classify a person.

Raw object detection rates for the uncontrolled experiment are presented in Table 4.8. Sensors have different FOV, range, mounting location perspective, recall, and precision rates which all jointly contribute to the differences in raw detection rate. In general, increased resolution increases object detection rate; the 8k camera most closely matches the lidar coverage in front of Skynet. Object tracking and classification average rates for the various sensor configurations coupled with radar are reported in Table 4.9. The most interesting conclusion of Table 4.9 is the similarity in object tracking rates between the C-8k+R and L+R which implies that real-time high resolution imagery detection could provide a reliable fair-weather alternative to lidar-based tracking for environments with well-spaced objects. For low resolution camera runs, the car and person classification rates are significantly below the object tracking rates because radar, which has no classification information, is providing the vast majority of object measurements. Lidar also has low classification rates but higher overall object tracking rate due to its long sensor range and high recall rate. High resolution imagery provides both increased object tracking rates and increased classification rates as the camera has good recall and excellent classification precision. As expected, coupling 8k imagery with lidar and radar provides the highest overall object detection and classification rates for tracked objects; interestingly the camera and lidar provide complementary information when fused. Later in the paper, sensor performance is evaluated for adverse weather and quantitative tracking positional accuracy, both important for closely-spaced crowded environments. For quantitative evaluations, 2k imagery resolution and a front

semi-circular sensor mask of 20m was used in order to minimize the differences between lidar and camera FOV and range.

Results from experiment trials evaluated with a complete sensor set C+L+R, and reduced sensor sets C+R, or L+R, across all four weather condition categories are summarized in Tables 4.10-4.11. Radar is included in both camera and lidar sensor set evaluations to improve estimation speed and robustness, and because, most road-worthy AGVs contain radar. The radar coverage on the front of the vehicle is very sparse and cannot detect pedestrians so radar only-tracking was not performed.

Table 4.10: Pedestrian Tracking Performance.

Sensors Used	**Object Tracked**	**Range Error (m)**	**Bearing Error (rad)**	**Correct Classification**	**Mis-Classification**	**Number of Returns**
C + L + R	0.995	1.230	0.081	0.864	0.018	3964
C + R	0.638	1.482	0.169	0.962	0.003	373
L + R	0.995	1.252	0.084	0.852	0.024	3972

Table 4.11: Vehicle Tracking Performance

Sensors Used	**Object Tracked**	**Range Error (m)**	**Bearing Error (rad)**	**Correct Classification**	**Mis-Classification**	**Number of Returns**
C + L + R	0.986	1.477	0.058	0.811	0.000	2594
C/H + R	0.655	1.578	0.110	0.818	0.000	1369
C + R	0.818	1.531	0.102	0.875	0.000	1776
L + R	0.982	1.410	0.053	0.509	0.000	2565

Table 4.11 also includes the case where heading detections are directly measured from the camera, labeled as C/H+R. In C/H+R, heading non-Gaussian

measurement ambiguity was poorly captured with a large Gaussian covariance on the measurement. Camera detections are also evaluated using the multiple hypothesis heading direction classifier shown in Equation (4.14) which split the heading measurement as a continuous angle to the vehicle length axis and discrete direction along the line; the heading multiple hypothesis method is labeled C+R. As shown in Table 4.11, having an accurate measure of the angle to the vehicle length axis proved useful and more accurate: C+R had more returns, more objects tracked, lower range and bearing errors and better classification rates than C/H+R. Properly orienting the vehicle heading significantly improved association and object state estimate for wheeled dynamics. All other analyses involving C+R in this paper utilized the multiple hypothesis heading measurement split method.

The downsampled camera sensor images reduced the number of car detections, resulting in a lower object tracking time fraction in C+R compared to L+R. The range and bearing errors were also larger for the C+R case compared to L+R. The full resolution image processing could improve the object tracking time fraction. The joint classification results from Tables 4.10-4.11 show that the C+R case has significantly better classification performance for cars compared to L+R, but for pedestrians, the improvement is less pronounced. This trend corresponds with expectation that the raw sensor classification accuracy difference between camera and lidar is larger for cars than pedestrians. The combined performance of the C+L+R case across both pedestrians and cars shows a combination of decreased range and bearing errors compared to C+R and increased classification performance compared to L+R. Intuitively, the benefits of having all three sensors is clearly shown in the C+L+R case: lidar is excellent at detecting objects and metrical information, whereas the camera is excellent at

classification. The joint fusion of all sensors achieves a much more accurate and robust solution.

4.4.6 Controlled Experiments: Weather Conditions in Joint Data Association, Tracking, and Classification

The car and pedestrian tracking results are combined to analyze the performance of the three sensor combinations across each weather condition category. Table 4.12 contains the combined L+R performance. All weather conditions had similar object tracking rates. Classification rates were highest for Night and second best during Cloudy & Wet conditions. Darker conditions from night and to a lesser extent from clouds provided reduced solar radiation noise for which the sensor had to contend. Surprisingly, the Cloudy & Wet conditions provided better lidar performance than daytime and best overall range estimates. Wet objects tend to scatter lidar returns; one plausible explanation is that the dry Suburban and dry pedestrians provided improved reflected signal returns compared to the reduced background noise returns. In summary, precipitation conditions degraded lidar performance most drastically in classification but also in range and bearing, while lidar performance improved in darker conditions because there was less reflected solar radiation to interfere with the lidar.

Table 4.13 contains the combined C+R performance. Similar to results shown in Tables 4.10-4.11, C+R object track rates in individual weather categories were all lower than the L+R weather categories because the downsampled camera sensor images reduced the number of car detections. Night had the worst object tracking fraction due to poor lighting conditions. Cloudy & Wet conditions

Table 4.12: Lidar + Radar (L+R) Performance in Weather

Description	Object Tracked	Range Error (m)	Bearing Error (rad)	Correct Classification	Mis-Classification	Number of Returns
Cloudy & Wet	0.988	0.700	0.063	0.726	0.000	1493
Sunny	0.999	1.437	0.083	0.664	0.000	1480
Night	0.987	1.424	0.060	0.872	0.000	1781
Snow & Rain	0.986	1.484	0.085	0.601	0.052	1783

provided best tracking fraction due to the uniform diffuse lighting conditions. Sunny daytime conditions have more glare and stark shadows to contend with than Cloudy conditions, resulting in worst range and bearing errors. One unexpected result was Night range and bearing error were less than Sunny conditions; it is hypothesized that this is due to shadow ambiguities in estimating an object bounding box. As expected, Snow & Rain precipitation degraded the object tracking fraction and also degraded range and bearing estimates. Surprisingly, classification rates were similar for Cloudy & Wet, Sunny, and Snow & Rain. The degraded Night classification, due to poor lighting conditions, was worse than Night classification in the L+R case. In summary, bad lighting conditions, especially at night, were more detrimental to camera performance than weather conditions.

Table 4.13: Camera + Radar (C+R) Performance in Weather

Description	Object Tracked	Range Error (m)	Bearing Error (rad)	Correct Classification	Mis-Classification	Number of Returns
Cloudy & Wet	0.916	1.414	0.083	0.971	0.000	552
Sunny	0.867	1.657	0.121	0.971	0.002	588
Night	0.620	1.425	0.112	0.590	0.000	432
Snow & Rain	0.739	1.627	0.127	0.951	0.000	450

Table 4.14 contains the complete sensor set C+L+R performance. Across all weather conditions the tracked object fraction was $\tilde{9}9\%$. Range and bearing errors are less than the C+R sensor set and correct classification is improved over the L+R set which is consistent with the controlled experiment results of Tables 4.10-4.11. As in the L+R sensor set, the C+L+R Cloudy & Wet conditions provided the best overall range estimates. The number of returns for L+R is three to four times that of C+R. The addition of camera data improved classification rates across all weather scenarios in C+L+R compared to L+R. However, due to the lower total number of camera returns, there were many sections tracked by L+R for which no camera information was available; thus the C+L+R classification rates are lower than that of C+R because of a discrepancy in the number of returns. Given a similar number of C+R and L+R returns, it is expected that the C+L+R classification rate would match or exceed that of the C+R. The mis-classification rate is nearly zero for all presented examples; the probabilistic filter is combining raw sensor classification information in an unbiased manner.

Table 4.14: Full Sensor Set (C+L+R) Performance in Weather

Description	Object Tracked	Range Error (m)	Bearing Error (rad)	Correct Classification	Mis-Classification	Number of Returns
Cloudy & Wet	0.985	0.825	0.056	0.876	0.000	1517
Sunny	0.998	1.544	0.077	0.875	0.000	1455
Night	0.989	1.320	0.070	0.889	0.000	1779
Snow & Rain	0.995	1.499	0.085	0.744	0.040	1807

The presented results demonstrate performance of a state-of-the-art camera detector and lidar configuration supplemented with radar exhibiting reliable performance across varying weather scenarios. Sensor diversity and a proba-

bilistic filter are critical for adding robustness to performance across weather scenarios.

4.4.7 Urban Driving Experiments: Qualitative Discussion of Performance in Weather Conditions

Experiments were conducted through Collegetown near Cornell's campus and downtown Ithaca to study performance in real-world scenarios involving diverse vehicle and pedestrian traffic on typical busy streets in various weather scenarios. Skynet was driven in each weather condition, Snow, Rain, Sunny, Cloudy, Wet & Humid, and Night, for over 30 miles and 2 hours of time, at speeds up to 35 miles per hour, in an assortment of urban conditions including two-lane one-way roads, two-lane two-way roads, assorted intersections involving pedestrians, and vehicles controlled with lights and signs, and around a number of building and neighborhood styles including downtown businesses, housing, and commercial box-store areas.

Observations made about overall sensor performance in clear weather are as follows: radar has few false positives, lidar is highly accurate at depth measurements, camera is highly accurate at correctly classifying vehicle and pedestrian detections, albeit with dependency on processing resolution; camera detection range is also dependent on processing resolution.

Active and passive air-borne disturbances are present during and after precipitation, such as rain or snow and during heavy fog. Water droplets and snow flakes could be detected in lidar and were visible in the camera frame. As has

been published elsewhere [98], [99], precipitation generally did not affect radars in a noticeable way in the experiments. The Ibeo XT lidar units have onboard rain drop filtering that examines the intensity and time response of individual lidar returns; if a double return is detected and the first response is lower in amplitude or shorter in duration, the first return is assumed to be a rain drop and the second return is the actual object. In principle, filtering airborne disturbances at the sensor's receiver processor could be beneficial, but in practice only a small portion of the rain returns were negated and had little effect on filtering snow flake returns.

The occupancy grid used for safety and collision avoidance was also susceptible to weather conditions. As shown in Figure 4.16, the lidar can return measurements of snow which are in turn passed to the occupancy grid. Snowflakes white color, larger size, and slower dynamics can cause returns of multiple lidar beams. Unfortunately, these returns are indistinguishable from object returns in intensity, and, importantly, they are close in proximity to the vehicle, creating safety concerns. Instead of lidar, commercial cars use sonar and radar for object detection and avoidance because they are less susceptible to airborne precipitation. When considering all-weather driving, heavy reliance on lidar in AGV research could present problems to commercialization.

Water and snow blown behind other vehicles in front of Skynet also made it difficult to track objects with lidar. Figure 4.17 shows a snow example where dozens of phantom objects are being tracked when the car is clearly visible in the camera's frame of view. Radar was completely unaffected by the water and snow spray, while the camera was only affected when the density of spray was strong enough to significantly cloud the lens and hide the vehicle. In the lidar

returns, the trailing spray was clustered arbitrarily as person- or car-sized depending on size of spray pattern and caused the RBPF to birth phantom tracked objects which followed behind the car. Dozens of phantom objects occupied several car-lengths of space, trailing the car while the actual car object was not reliably tracked or classified. In Figure 4.17, the black arrow points to the true location of the car, but no tracked object is estimated in the car's true location.

The most surprising airborne phenomenon was exhaust plumes, which can cause lidar returns. During cool conditions, vehicle exhaust or steam venting from city street tunnels into ambient temperatures with a low dew-point caused the exhaust to condense into clouds and create large plumes that created lidar returns. Interestingly, exhaust and steam plumes that are barely visible to the human eye or cameras in the optical spectrum can cause significant lidar returns. Figure 4.18 shows an example of lidar returns at two time instances and the corresponding camera image; black arrows point to the exhaust plume location. Initially, the plume was clustered with the vehicle, but as the vehicle drove through the green light, the plume was blown behind the vehicle and clustered as both a person-sized and car-sized object. This phenomenon was most prevalent when driving in winter, cool, or rainy conditions and largely non-existent when ambient temperatures were warm and the dew-point was low.

Sensor fouling was most common in snow and rain, but can also occur in dusty conditions. Wipers, such as those typically found on windshields, are important for all sensors; given enough sensor surface accumulation of snow, ice, or dirt, radar, camera, and lidar sensors malfunction. Radar is robust to water surface accumulation on the sensor but snow pack and ice accumulation from highway driving can disable radar. Camera and lidar sensors are both sensitive

to any sensor surface accumulation that blocks light; the lidar could operate with a wet lens cover; the camera scenes of pedestrians and cars through a wet lens became unrecognizable to the detector. The Velodyne lidar naturally stays clean, given its rotation which limits water and snow from directly hitting the lens. What water does hit the lens tends to blow off from wind and centrifugal force.

Environmental surface accumulation has the potential to cause a variety of unexpected sensor behaviors. For lidar, wet object surfaces decreased return rates while snowy object surfaces increased return rates. The camera was unaffected in general by light environmental surface accumulation, but under heavy snow the camera eventually was unable to distinguish edges of objects and their environment. Figure 4.19 shows an example camera detection of a snow covered car.

Reflections were noticed in the camera when driving by shiny buildings, as captured in Figure 4.19. This example poses less of a safety issue in that its validity could be reasoned about using the detected location of the obstacle. Selecting an exposure setting for the cameras was somewhat challenging. For simplicity, a uniform exposure across lenses was selected to support easy panorama creation for detection processing. However, uniform exposure creates problems for areas where lighting has gross variations, such as dusk, dawn, and oncoming headlights. These lighting problems are magnified with a wet road surface due to reflections.

It has been reported [116] that radar can have problems with shiny and reflective glass or objects, but this was not observed in Skynet recorded data.

Multi-path lidar was a significant problem on wet surfaces, as shown in the examples in Figure 4.20. In airborne topology mapping, [117] multi-path lidar has been reported and is typically avoided by controlling the inclination angle to ground targets to be nearly vertical. However, smooth wet roads cause lidar to reflect on its way to or from the target. The lidar collector which receives the return, determines the angle corresponding to the range measurement. If the lidar beam initially reflected off the ground, and then reflected off an object before returning to the collector, the object is projected to be farther away than it actually is due to the longer round trip path. Over-estimating the distance to a target can be dangerous and lead to AGV collisions. Measurement gating might offer some potential ways to alleviate the multi-path problem.

If the lidar beam first reflects off the object, and then reflects off the ground on the return path, the range to the object is projected to go through the ground. Many AGVs also estimate the location of the ground or roadways with some form of ground plane detection; examples of multi-path lidar returns projected to have originated from below ground level, as defined by Skynet's ground plane detection, are shown in Figure 4.21. Due to lidar scattering off the wet road surface creating multi-path returns, the ground plane was estimated below the actual ground level shown in Figure 4.21 left. For comparison, Figure 4.21 right shows an example of a typical ground plane estimate in dry conditions with dense lidar returns. Low ground plane estimation could lead to over-estimating object range calculations for camera detections, which could lead to AGV collisions.

From the above experiments, in both fair weather and not, camera, lidar, and radar sensors were found to compliment each other. Lidar showed improved

returns from snow covered objects and excellent night performance. However, lidar challenges included multipath reflections on rainy roads, difficulty with ground plane estimation in wet conditions, clustering and tracking issues with snow including that blown up by other traffic or wind, response degradation to wet vehicles, response to exhaust plumes in cold and wet conditions, and errant occupancy grid response. The camera was more robust to precipitation, and provided the most accurate classifications, but struggled with dark conditions or lighting variations within a single scene. Radar was most robust to different weather conditions and provides accurate velocity information, but cannot detect pedestrians and lacks some of the depth, shape, and size accuracy of lidar or the classification accuracy of the camera. Occasional clutter also obscures radar returns for distant objects or closely spaced objects at similar distances and speeds.

4.5 Conclusion

A novel real-time probabilistic joint data association, tracking, and classification system for an autonomous ground vehicle is formulated. Additionally, a state-of-the-art vision detection algorithm that includes heading detection for autonomous ground vehicle applications is integrated and compared. With the incorporation of lidar clustering, radar sensors, and pose, a real-time demonstration of the joint probabilistic perception algorithm was conducted in varying weather conditions and using different subsets of sensors. Monte Carlo simulations, repeatable controlled experiments, and a lengthy real-world urban data collection demonstrated performance and identified new challenges with weather perception and unique capabilities of a joint association, tracking, and

classification solution.

Many observations were made regarding autonomous ground vehicle performance in weather. In general, lidar was most brittle to laser blockages, multipath returns, airborne precipitation, and wet surfaces, and most robust to lighting conditions, while providing object shape and size information. The camera was most brittle to dark lighting conditions and glare, but was more robust to precipitation than lidar. Glare, which was often present in wet conditions from headlights or sunshine, would reflect brightly off road, vehicle, and building surfaces, making camera exposure selection difficult and degrading object detection performance. Radar has the best robustness in performance to all weather conditions, but often cannot detect pedestrians and provides less information about object shape, size, and classification than lidar or camera.

Given the various limitations of existing sensors, there is much opportunity for future development of sensor hardware, sensor data processing, and perception algorithm advancements. Cheap and reliable lens cleaning for cameras and to a lesser degree for lidar and radar are necessary for reliable operation in any form of precipitation. Improving the dynamic range of cameras, composing high dynamic range images, or actively modulating exposure across the CCD may provide some potential for improving camera operation at night and during high glare conditions. Extending tracking of precipitation [118] with classification to categorize various weather phenomenon could improve individual sensor performance or perception system performance. As computing power increases and image detection methods advance, higher resolution image processing could enable higher detection rates and longer detection ranges for the vision. Direct estimation of the current weather condition could al-

low active of toggling sensors, falling back to a reduced base sensor set, or weighting sensor returns if each sensors performance is accurately characterized across weather conditions; for example, spurious lidar returns of snow, exhaust plumes, or phantom occupancy grid objects could be automatically discarded if the weather condition and sensor's weather sensitivities were known by the vehicle. In summary, sensor diversity and joint estimation of data association, tracking, and classification proved beneficial towards robust performance in all-weather conditions.

Acknowledgments

Authors would like to thank Trimble for providing Omnnistar HP DGPS corrections service, NVIDIA for providing a GTX980 GPU, Kevin Wyffels for insightful discussions on classification formulation, and You Won Park for constructing the Ladybug3 Camera Mount.

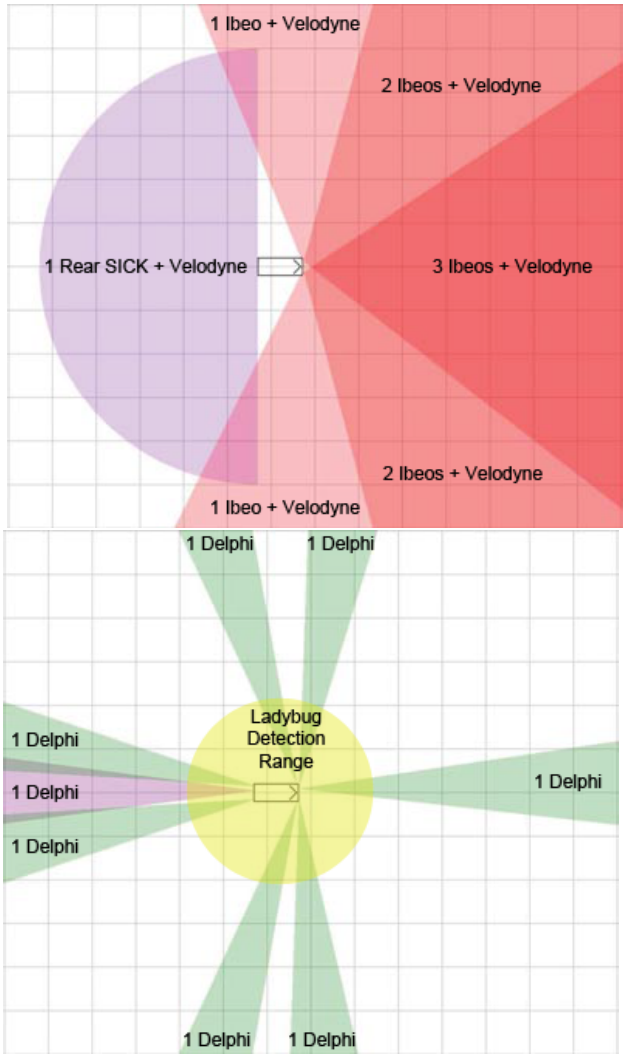

Figure 4.6: Sensor placement and coverage for lidar (left), camera (right), and radar (right) sensors. Skynet sits in the center of the diagram, facing right. Radar coverage is somewhat sparse in the forward direction. Original placement was chosen to detect specific direction: forward for oncoming cars in opposing lanes, left and right for merging or intersection situations, and rear for passing situations. The Lidar coverage is most dense in front of skynet. The Ladybug camera imagery detection range depends on resolution used for frame processing.

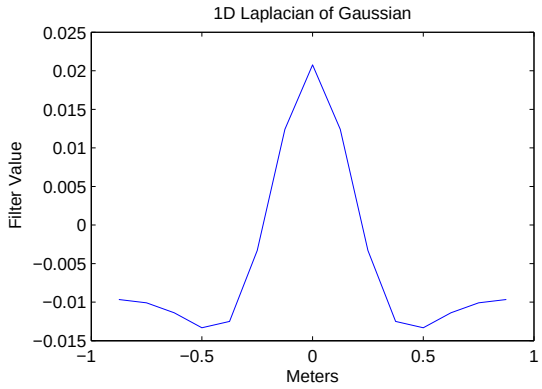

Figure 4.7: 1D LoG filter for person lidar clustering.

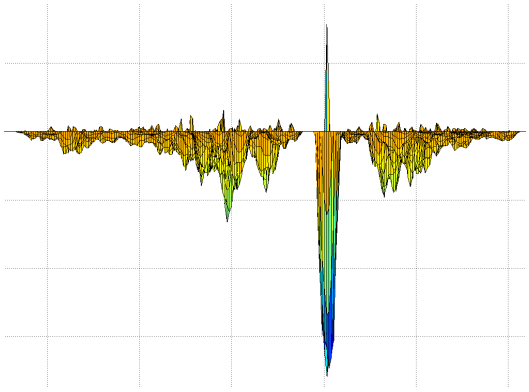

Figure 4.8: Example LoG filter response to lidar returns from an uncluttered roadway scene with one person.

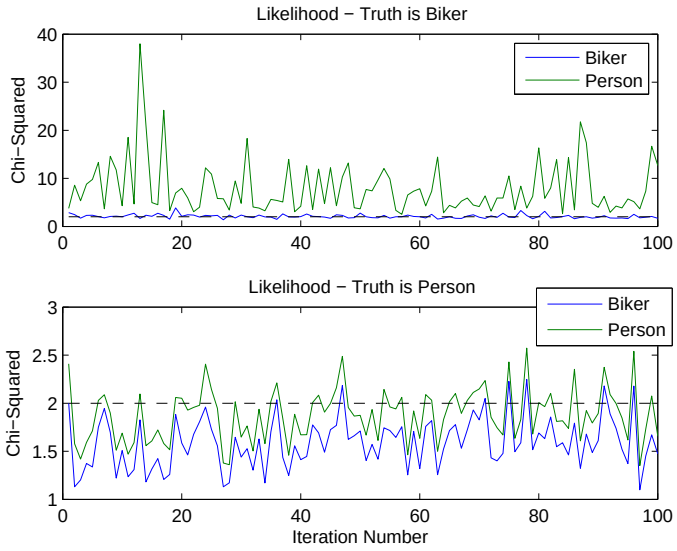

Figure 4.9: Monte Carlo simulation results. Black dashed line represents ideal mean value for true classification.

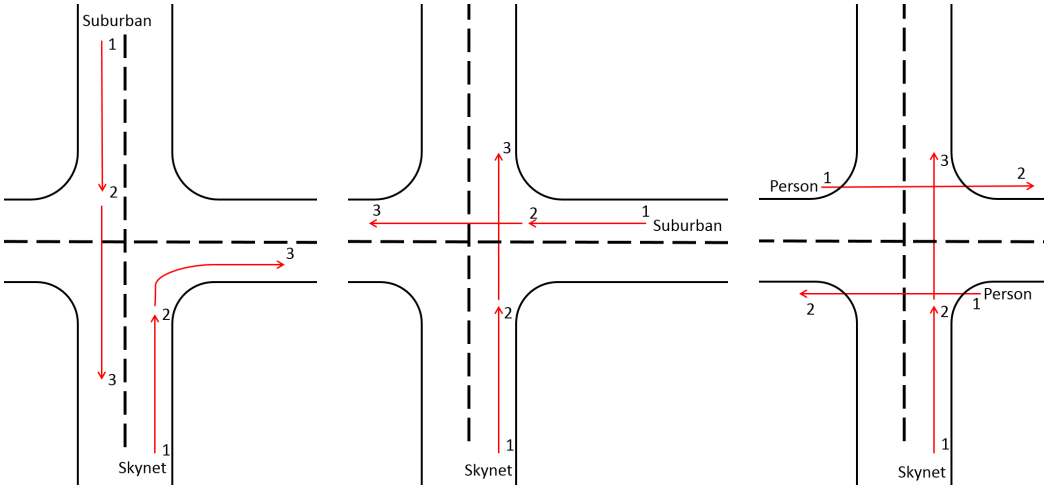

Figure 4.10: Three intersection scenarios used in the experiments are defined as shown.

Figure 4.11: Visual data and example detections for the four weather condition categories during the intersection experiments. Rows top to bottom: Cloudy & Wet, Sunny, Night, Snow & Rain.

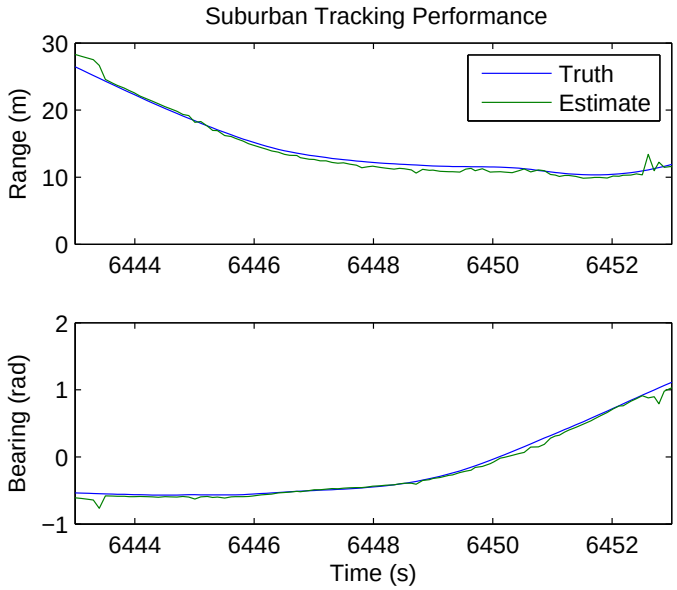

Figure 4.12: Example truth and estimated object track for Suburban.

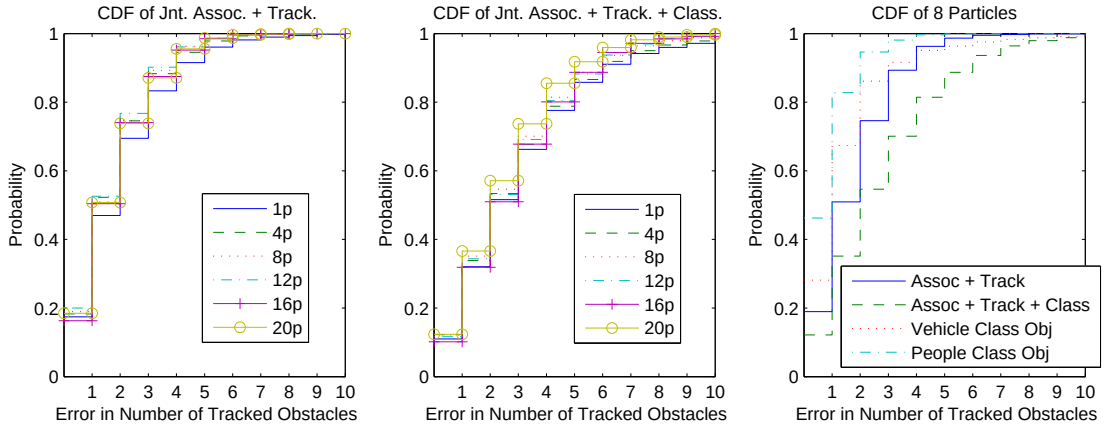

Figure 4.13: Comparison of number of particles selected for joint data association and tracking versus joint data association, tracking, and classification. Figures plot cumulative distribution function (CDF) of errors in number of tracked objects for various particle counts versus a truth simulation using 50 particles. The right-most plot compares how the addition of classification affects the overall object error CDF for 8 particles. The added complexity, increased number of measurements, and addition of classification decreases the overall CDF, implying more particles are necessary for similar performance when adding classification. However, CDF errors for car and person classified objects is decreased, that is, classification improved the object track consistency for classified objects compared to omitting classification.

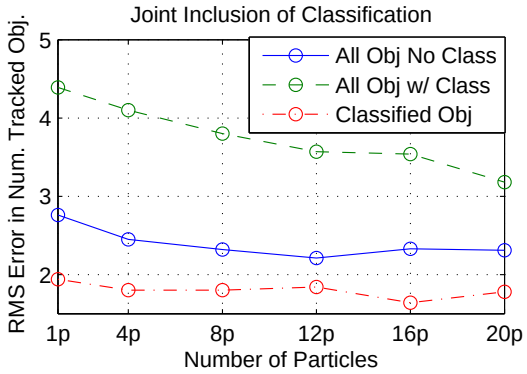

Figure 4.14: Comparison of overall errors in number of tracked objects. This graphic plots data from Table 4.5.

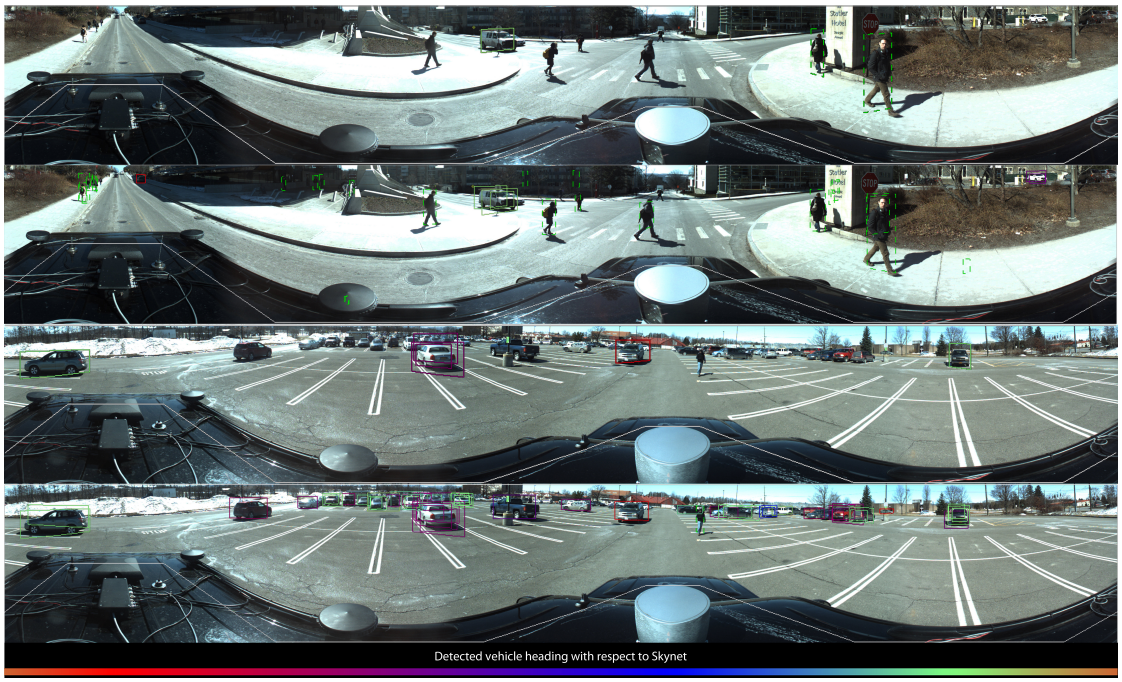

Figure 4.15: Example of detection performance of the car and person detector for the same images but different processing resolutions. The second and fourth photos show detections for 8k resolution imagery from Ladybug camera. The first and third photos show car and person detections for 2k imagery used for remaining experiments.

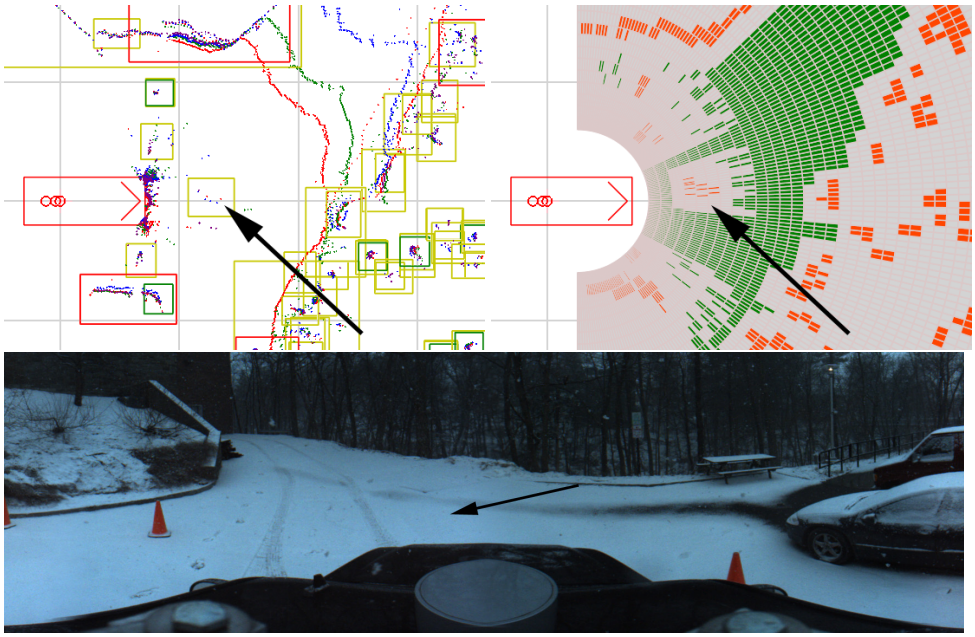

Figure 4.16: Experimental time snapshot showing lidar returns from snow flakes. In top-down depictions, Skynet is facing right. Upper left: lidar and clustering. Upper right: occupancy grid. Bottom: view of scene. The black arrow point to the same location, and denote lidar sensor returns and the resulting occupancy in the occupancy grid due to detecting snow flakes as objects.

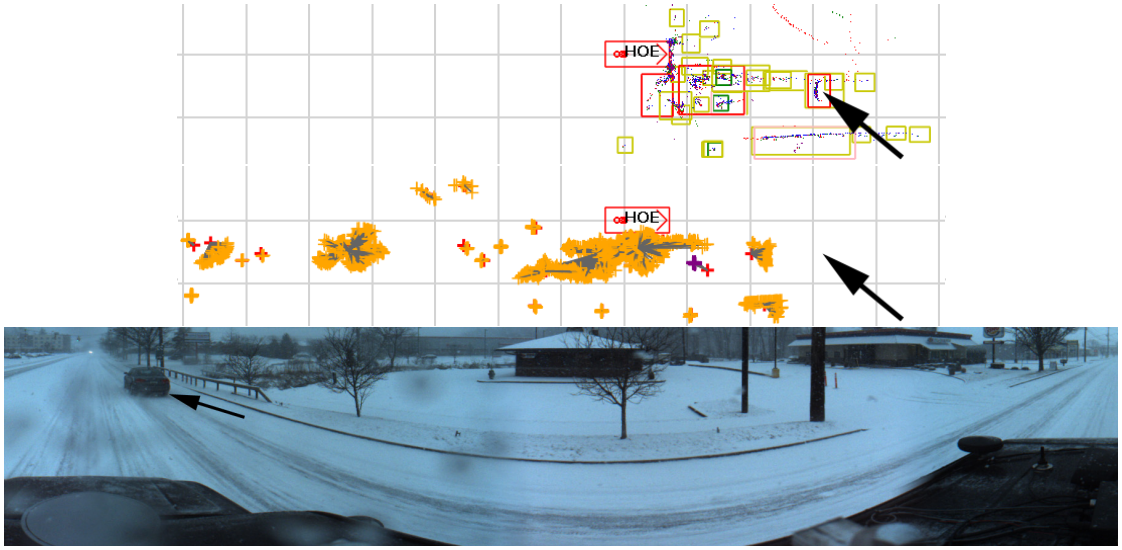

Figure 4.17: Lidar returns of snow trailing behind vehicle causes errant clustering and therefore errant tracked objects. Arrows point to a vehicle object location. Top: lidar returns and clustering. Middle: tracked objects. Bottom: Skynet faces left and shows clear road.

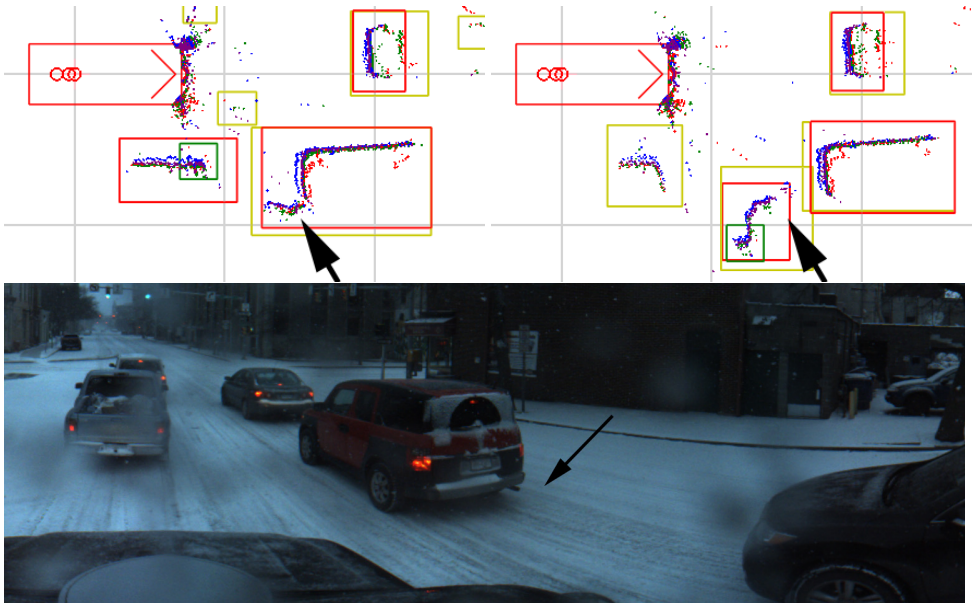

Figure 4.18: Example of an exhaust plume causing lidar sensor returns. Upper left: lidar and clustering. Right: same scene 2 seconds later as vehicle moves through green light. Bottom: view of scene. Black arrow points to example of optically invisible plume being detected by lidar and clustered as a person (green box) and car (red box).

Figure 4.19: Left: detection of car in snow. Right: detection of car in store window reflection.

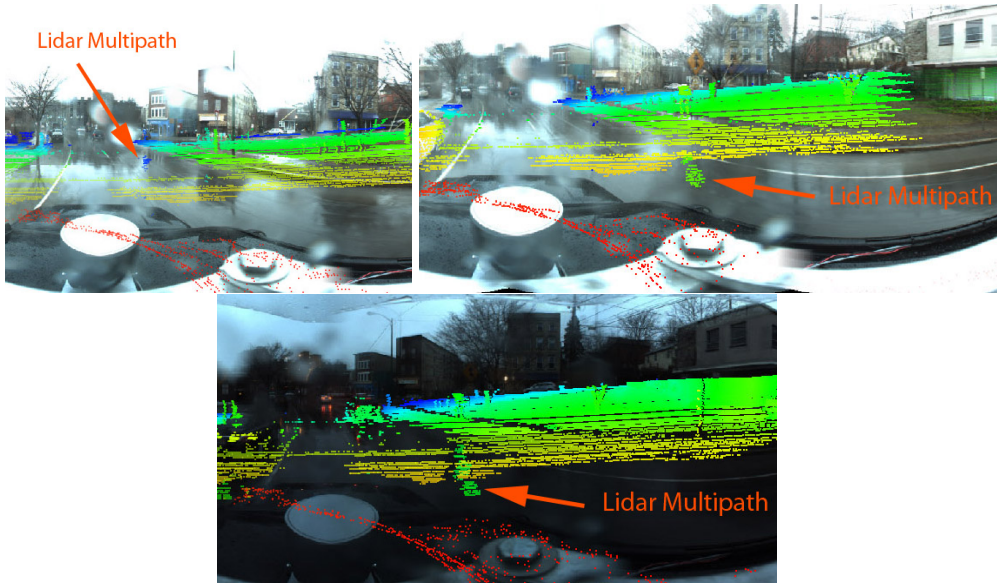

Figure 4.20: Examples of lidar multipath from wet conditions.

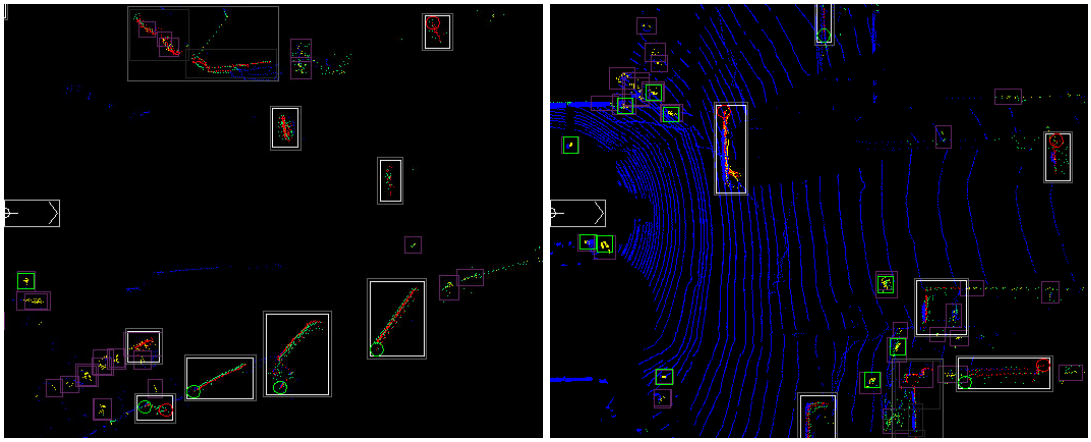

Figure 4.21: Ground view of lidar returns inside the lidar clustering routine. Blue dots are classified as ground plane; white boxes are car-sized objects; green boxes are person-sized objects. Left: Ground plane estimated below actual ground level from sparse multipath returns in wet conditions. Right: Typical ground plane estimate in dry conditions.

CHAPTER 5

CONCLUSION

The research presented has demonstrated advancements in thermal model acquisition for buildings and robust all-weather perception for autonomous ground vehicles. These advancements can enable new model predictive control capabilities for reduced energy consumption and improved occupant comfort in buildings and enable autonomous vehicles to operate in a wider range of common environmental conditions.

In chapter 2, a multi-mode implementation of a multi-zone UKF was presented as a scalable and rapidly deployable system identification routine for building thermal dynamics. Using a passive 5-room model, the UKF demonstrated the ability to learn both dynamics parameters for a thermal network and unknown disturbances. 24-Hour predictions from UKF estimated parameters yielded accurate results which were validated with EnergyPlus simulations using a full year of data. The UKF, a data-driven, model-based approach, amenable to augmentation with numerical methods, provides a promising step towards a scalable framework to realize advanced BAS predictive controllers.

In chapter 3, MPC used an online learned model to outperform a well-tuned thermostat controller and demonstrate the potential that online autonomous methods have in building systems. Various online methods were presented for excitation, and the effects of excitation on the estimation filter were analyzed. By coupling gray-box estimation, predictive control algorithms, online self-excitation, and monitoring into a single framework, the potential has been highlighted for low-cost scalable methods to save energy and improve occupant comfort in buildings.

In chapter 4, a novel real-time joint probabilistic data association, tracking, and classification system for an AGV was formulated. Additionally a state-of-the-art vision detection algorithm that includes heading information for AGV applications was implemented. With the incorporation of lidar clustering, radar sensors, and pose, a real-time demonstration of the joint probabilistic perception algorithm was conducted in varying weather conditions. Repeatable controlled experiments plus real-world urban data collection identified new challenges with weather perception and unique capabilities of a joint probabilistic perception solution.

Many observations were made regarding AGV performance in weather. In general, lidar provided highly detailed object shape and size information, was most brittle to precipitation and wet surfaces, and was most robust to lighting conditions. The camera provided accurate classification and location information, was most brittle to dark lighting conditions and glare, but had more robustness to precipitation than lidar. Radar had the best robustness in performance to all weather conditions but provided less information about object shape, size, and classification than lidar or camera. The joint association, tracking, and classification algorithm was implemented in a real-time framework that demonstrated robust all-weather perception performance.

This dissertation has made the following novel contributions for HVAC control systems in buildings:

- Literature survey on control-oriented thermal modeling for buildings;
- Development of minimal parameterization for dynamics estimation;
- Generalization of estimation of thermal disturbance patterns;

- Explanation of a multi-mode heuristic for simultaneous parameter and disturbance estimation;
- Development of an Experiment Generator that determines, based on the current model, which zones should be excited;
- Development of an Experiment Selector that automatically selects and runs experiments while obeying energy and occupant constraints;
- Observability analysis confirming and explaining the basis for the excitation approach;
- Demonstration of simultaneous thermal model and disturbance estimation and corresponding improved HVAC control with MPC;

and the following novel contributions for autonomous ground vehicle perception:

- Demonstration of object classification in an urban environment based on multiple model tracking;
- Formulation and demonstration of a real-time joint probabilistic method to solve data association, tracking, and classification for an AGV roadway environment;
- Examination of if and how state-of-the-art vision algorithms can compliment or replace lidar and radar sensors;
- Investigation of sensor and perception algorithm performance in adverse weather and lighting conditions;
- Demonstration of robust all-weather perception using formulated joint Bayesian perception algorithm.

BIBLIOGRAPHY

[1] P. Radecki and B. Hencey, "Online building thermal parameter estimation via unscented kalman filtering," (Montreal, Canada,), American Controls Conferece, 2012.

[2] P. Radecki and B. Hencey, "Online thermal estimation, control, and self-excitation of buildings," (Florence, IT,), IEEE Conference on Decision and Control, 2013.

[3] R. Brown, "U . S . Building-Sector Energy Efficiency Potential," tech. rep., Lawrence Berkeley National Laboratory, Berkeley, CA,, 2008.

[4] T. Y. Chen and A. K. Athienitis, "Investigation of practical issues in building thermal parameter estimation," *Building and Environment*, vol. 38, pp. 1027–1038, Aug 2003.

[5] X. Li and J. Wen, "Review of building energy modeling for control and operation," *Renewable and Sustainable Energy Reviews*, vol. 37, pp. 517–537, 2014.

[6] V. M. Zavala, D. Skow, T. Celinski, and P. Dickinson, "Techno-economic evaluation of a next-generation building energy management system," tech. rep., 2011.

[7] Z. O'Neill and S. Narayanan, "Model-based estimation of cold room temperatures in a supermarket refrigeration system," *Applied Thermal Engineering*, vol. 73, pp. 819–830, 2014.

[8] O. Ogunsola and L. Song, "Application of a simplified thermal network model for real-time thermal load estimation," *Energy and Buildings*, vol. 96, pp. 309–318, 2015.

[9] Y. Lin, T. Middelkoop, and P. Barooah, "Issues in identification of control-oriented thermal models of zones in multi-zone buildings," (Hawaii,), IEEE Conference on Decision and Control, 2012.

[10] H. xiang Zhao and F. Magoules, "A review on the prediction of building energy consumption," *Renewable and Sustainable Energy Reviews*, vol. 16, pp. 3586–3592, 2012.

[11] I. Hazyuk, C. Ghiaus, and D. Penhouet, "Optimal temperature control of intermittently heated buildings using model predictive control: Part i - building modeling," *Building and Environment*, vol. 51, pp. 379–387, 2011.

[12] "NSF CMMI Workshop on Building Systems," tech. rep., UIUC, Urbana, IL,, 2010.

[13] M. Trcka and J. L. Hensen, "Overview of hvac system simulation," *Automation in Construction*, vol. 19, pp. 93–99, Mar 2010.

[14] D. Kim and J. E. Braun, "A general approach for generating reduced-order models for large multi-zone buildings," *Journal of Building Performance Simulation*, vol. 8, no. 6, pp. 435–448, 2015.

[15] J. Cai and J. E. Braun, "Efficient and robust training methodology for inverse building modeling and its application to a multi-zone case study," Efficient and Robust Training Methodology for Inverse Building Modeling and Its Application to a Multi-zone Case Study, 2012.

[16] S. Wang and X. Xinhua, "Parameter estimation of internal thermal mass of building dynamic models using genetic algorithm," *Energy Conversion and Management*, vol. 47, pp. 1927–1941, Aug 2006.

[17] S. Karatasou, M. Santamouris, and V. Geros, "Modeling and predicting building's energy use with artificial neural networks: Methods and results," *Energy and Buildings*, vol. 38, pp. 949–958, Aug 2006.

[18] S. F. Fux, A. Ashouri, M. J. Benz, and L. Guzzella, "Ekf based self-adaptive thermal model for a passive house," *Energy and Buildings*, vol. 68, pp. 811–817, 2014.

[19] M. Maasoumy, B. Moridian, M. Razmara, M. Shahbakhti, and A. Sangiovani-Vincentelli, "Online simultaneous state estimation and parameter adaptation for building predictive control," (Stanford, CA,), Dynamic System and Control Conference, 2013.

[20] A. Martincevic, A. Starcic, and M. Vasak, "Parameter estimation for low-order models of complex buildings," (Istanbul,), 5th IEEE PES Innovative Smart Grid Technologies Europe, 2014.

[21] B. Huchuk, C. A. Cruickshank, W. O'Brein, and H. Gunay, "Recursive

thermal building model training using ensemble kalman filters," (Ottowa, Canada,), eSim, 2014.

[22] Y. Ma, A. Kelman, A. Daly, and F. Borrelli, "Predictive control for energy efficient buildings with thermal storage: Modeling, simulation, and experiments," *Control Systems Magazine, IEEE*, vol. 32, 2012.

[23] M. Wetter and C. Haugstetter, "Modelica versus trnsys - a comparison between an equation-based and a procedural modeling language for building energy simulation," (Cambridge, MA,), Proceedings of SimBuild, 2nd National Conference of IBPSA-USA, International Building Performance Simulation Association, 2006.

[24] S. Goyal and P. Barooah, "A method for model-reduction of nonlinear building thermal dynamics," in *American Control Conference (ACC), 2011*, pp. 2077–2082, 2011.

[25] K. Deng, P. Barooah, P. G. Mehta, and S. P. Meyn, "Building thermal model reduction via aggregation of states," (Baltimore, MD,), pp. 5118–5123, American Control Conference, IEEE, 2010.

[26] N. L. Jones and D. P. Greenberg, "Fast computation of incident solar radiation from preliminary to final building design," (Sydney, Australia,), 12th International Conference of the International Building Performance Simulation Association, 2011.

[27] S.-K. Lin, "Minimal linear combinations of the inertia parameters of a manipulator," *IEEE Transactions on Robotics and Automation*, vol. 11, no. 3, pp. 360–373, 1995.

[28] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. 2001.

[29] D. Greenberg, K. Pratt, B. Hencey, N. Jones, L. Schumann, J. Dobbs, Z. Dong, D. Bosworth, and B. Walter, "Sustain: An experimental test bed for building energy simulation," *Energy and Buildings*, vol. 58, pp. 44–57, 2013.

[30] "Ashrae standard 55, thermal environmental conditions for human occupancy," tech. rep., 2010.

[31] J. Dobbs and B. Hencey, "A comparison of thermal zone aggregation methods," (Hawaii,), 51st IEEE Conference on Decision and Control, 2012.

[32] S. Privara, J. Cigler, Z. Vana, F. Oldewurtel, C. Sagerschnig, and E. Zacekova, "Building modeling as a crucial part for building predictive control," *Energy and Buildings*, vol. 56, pp. 8 – 22, 2013.

[33] S. Bengea, V. Adetola, K. Kang, M. J. Liba, D. Vrabie, R. Bitmead, and S. Narayanan, "Parameter estimation of a building system model and impact of estimation error on closed-loop performance," (Orlando, FL,), 50th IEEE Conferenceon Decision and Control and European Control Conference, 2011.

[34] A. Aswani, N. Master, J. Taneja, V. Smith, A. Krioukov, D. Culler, and C. Tomlin, "Identifying models of hvac systems using semiparametric regression," (Monteal,), American Control Conference, 2012.

[35] P. A. Ioannou and J. Sun, *Robust adaptive control*. Courier Corporation, 2012.

[36] I. Hazyuk, C. Ghiaus, and D. Penhouet, "Model predictive control of thermal comfort as a benchmark for controller performance," *Automation in Construction*, vol. 43, pp. 98–109, 2014.

[37] H. B. Gunay, J. Bursill, B. Huchuk, W. O'Brein, and I. Beausoleil-Morrison, "Shortest-prediction-horizon model-based predictive control for individual offices," *Buidling and Environment*, vol. 82, pp. 408–419, 2014.

[38] C. R. Touretzky and R. Patil, "Building-level power demand forecasting framework using building specific inputs: development and applications," *Applied Energy*, vol. 147, pp. 466–477, 2015.

[39] G. Bianchini, M. Casini, A. Vicino, and D. Zarrilli, "Receding horizon control for demand-response operation of building heating systems," (Los Angeles, California,), 53rd IEEE Conference on Decision and Control, 2014.

[40] W. J. Cole, J. D. Rhodes, W. Gorman, K. X. Perez, M. E. Webber, and T. F. Edgar, "Community-scale residential air conditioning control for effective grid management," *Applied Energy*, vol. 130, pp. 428 – 436, 2014.

[41] J. D. Rhodes, W. H. Gorman, C. R. Upshaw, and M. E. Webber, "Using

{BEopt} (energyplus) with energy audits and surveys to predict actual residential energy usage," *Energy and Buildings*, vol. 86, pp. 808 – 816, 2015.

[42] S. Kiliccote, M. A. Piette, and D. Hansen, "Advanced controls and communications for demand response and energy efficiency in commercial buildings," No. 150, (Pittsburgh, PA,), Second Carnegie Mellon Conference in Electric Power Systems, 2006.

[43] M. Behl, T. Nghiem, and R. Mangharam, "Model-iq: Uncertainty propagation from sensing to modeling and control in buildings," (Berlin,), ACM/IEEE International Conference on Cyber-Physical Systems, 2014.

[44] M. Maasoumy, M. Razmara, M. Shabakhti, and A. Sangiovanni-Vincentelli, "Handling model uncertainty in model predictive control for energy efficient buildings," *Journal of Energy and Buildings*, vol. 77, pp. 377–392, 2014.

[45] G. Platt, J. Li, R. Li, G. Poulton, G. James, and J. Wall, "Adaptive hvac zone modeling for sustainable buildings," *Energy and Buildings*, vol. 42, pp. 412–421, Apr 2010.

[46] J. Wen and T. F. Smith, "Development and validation of online models with parameter estimation for a building zone with VAV system," *Energy and Buildings*, vol. 39, pp. 13–22, Jan. 2007.

[47] N. R. Kristensen, H. Madsen, and S. B. Jørgensen, "Parameter estimation in stochastic grey-box models," vol. 40, pp. 225–237, Feb. 2004.

[48] C. Agbi, Z. Song, and B. Krogh, "Parameter identifiability for multi-zone building models," (Maui, Hawaii,), 51st IEEE Conference on Decision and Control, 2012.

[49] D. Gorni and A. Visioli, "Practical issues in modelling the temperature for the control of smart buildings," European Workshop on Advanced Control and Diagnosis, 2014.

[50] W. Bernal, M. Behl, T. X. Nghiem, and R. Mangharam, "Mle+: A tool for integrated design and deployment of energy efficient building controls," (Toronto,), Buildsys '12, 2012.

[51] S. Privara, Z. Vana, D. Gyalistras, J. Cigler, C. Sagerschnig, M. Morari, and

L. Ferkl, "Modeling and identification of a large multi-zone office building," in *2011 IEEE International Conference on Control Applications (CCA)*, pp. 55–60, Sept. 2011.

[52] M. Guay and D. Burns, "A comparison of extremum seeking algorithms applied to vapor compression system optimization," in *American Control Conference (ACC), 2014*, pp. 1076–1081, June 2014.

[53] X. Bombois, G. Scorletti, M. Gevers, P. Van den Hof, and R. Hildebrand, "Least costly identification experiment for control," vol. 42, pp. 1651–1662, Oct. 2006.

[54] J. Crigler and S. Privara, "Subspace identification and model predictive control for buildings," (Singapore,), 11th Int. Conf. Control, Automation, Robotics, and Vision, 2010.

[55] S. Dasgupta, B. D. O. Anderson, and R. John Kaye, "Persistence of excitation conditions for partially known systems," vol. 30, pp. 547–550, Mar. 1994.

[56] C. Atkeson and J. Santamaria, "A comparison of direct and model-based reinforcement learning," in *, 1997 IEEE International Conference on Robotics and Automation, 1997. Proceedings*, vol. 4, pp. 3557–3564 vol.4, 1997.

[57] J. Cai, D. Kim, J. E. Braun, and J. Hu, "Optimizing zone temperature setpoint excitation to minimize training data for data-driven dynamic building models," American Controls Conference, 2016.

[58] G. Welch and G. Bishop, "An introduction to the kalman filter," tech. rep., Chapel Hill, NC,, 2003.

[59] S. J. Julier and J. K. Uhlmann, "A new extension of the kalman filter to nonlinear systems," (Orlando, FL,), SPIE: The Proceedings of AeroSense: The 11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls, SPIE, 1997.

[60] I. Reichert, P. Olney, T. Lahmer, and V. Zabel, "Comparison of Different Approaches for the Model-Based Design of Experiments," in *Model Validation and Uncertainty Quantification, Volume 3*, Conference Proceedings of the Society for Experimental Mechanics Series, pp. 135–141, Springer International Publishing, 2015.

[61] A. Bemporad and M. Morari, *Robustness in identification and control*, ch. Robust model predictive control: A survey, pp. 207–226. London: Springer London, 1999.

[62] J. Matusko and F. Borrelli, "Scenario-based approach to stochastic linear predictive control," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pp. 5194–5199, Dec 2012.

[63] E. C. Kerrigan and J. M. Maciejowski, "Soft constraints and exact penalty functions in model predictive control," (Cambridge, UK,), UKACC International Conference, 2000.

[64] M. Grant and S. Boyd, "Cvx: Matlab software for disciplined convex programming, version 2.0," 2013.

[65] M. Grant and S. Boyd, *Graph implementations for nonsmooth convex programs, Recent Advances in Learning and Control (a tribute to M. Vidyasagar), V. Blondel, S. Boyd, and H. Kimura, editors*, pp. 95–110. Springer-Verlag Limited, 2008.

[66] "Gurobi optimizer reference manual," 2015.

[67] J. B. Jorgensen, "Convex qp algorithms for linear mpc with soft output constraints," (University of Leuven, Flanders, Belgium,), OPTEC Workshop on Large Scale Quadratic Programming, 2010.

[68] B. A. Ogunnaike, "A contemporary industrial perspective on process control theory and practice," *Annual Reviews in Control*, vol. 20, pp. 1–8, 1996.

[69] K. Popper, *Conjectures and Refutations: The Growth of Scientific Knowledge*. 1963.

[70] H. Hjalmarsson, "From experiment design to closed-loop control," vol. 41, no. 3, pp. 393–438, 2005.

[71] R. Schenkendorf, A. Kremling, and M. Mangold, "Optimal experimental design with the sigma point method," *IET systems biology*, vol. 3, no. 1, pp. 10–23, 2009.

[72] M. L. Psiaki, F. Martel, and P. K. Pal, "Three-axis attitude determination via kalman filtering of magnetometer data," *Journal of Guidance, Control, and Dynamics*, vol. 13, no. 3, pp. 506–514, 1990.

[73] NHTSA, *Preliminary Statement of Policy Concerning Automated Vehicles*. National Highway Traffic Safety Administration, 2014.

[74] SAE, *Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems - J3016*. Society of Automotive Engineers: On-Road Automated Vehicle Standards Committee, 2013.

[75] E. Ackerman, "Korean Competition Shows Weather Still a Challenge for Autonomous Cars," in *IEEE Spectrum, Cars that Think Website*, 2014. http://spectrum.ieee.org/cars-that-think/transportation/advanced-cars/japan-competition-shows-weather-still-a-challenge-for-autonomous-cars.

[76] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society Conference on*, vol. 1, pp. 886–893 vol. 1, June 2005.

[77] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, pp. 1627–1645, Sept 2010.

[78] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *NIPS: Neural Information Processing Systems*, (Lake Tahoe, Nevada), 2012.

[79] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, 2014.

[80] R. Girshick, "Fast R-CNN," in *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[81] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *Neural Information Processing Systems (NIPS)*, Microsoft Research, 2015. http://arxiv.org/pdf/1506.01497v2.pdf.

[82] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition." http://arxiv.org/pdf/1512.03385v1.pdf, Dec 2015.

[83] L. Deng, "A Tutorial Survey of Architectures, Algo-

rithms, and Applications for Deep Learning," *APSIPA Transactions on Signal and Information Processing*, 2014. http://research.microsoft.com/apps/pubs/default.aspx?id=204048.

[84] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object Detection with Discriminatively Trained Part Based Models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[85] B. Douillard, J. Underwood, N. Kuntz, V. Vlaskine, A. Quadros, P. Morton, and A. Frenkel, "On the segmentation of 3D LIDAR point clouds," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 2798–2805, May 2011.

[86] D. Korchev, S. Cheng, Y. Owechko, and K. Kim, "On Real-Time LIDAR Data Segmentation and Classification," in *Proceedings of Image Processing, Computer Vision, & Pattern Recognition ICPV*, 2013.

[87] D. Held, J. Levinson, and S. Thrun, "Precision tracking with sparse 3D and dense color 2D data," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 1138–1145, May 2013.

[88] D. Held, J. Levinson, and S. Thrun, "A probabilistic framework for car detection in images using context and scale," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 1628–1634, May 2012.

[89] M. A. Sadeghi and D. Forsyth, "30Hz Object Detection with DPM V5," in *Computer Vision – ECCV 2014* (D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, eds.), vol. 8689 of *Lecture Notes in Computer Science*, pp. 65–79, Springer International Publishing, 2014.

[90] J. S. Huang, "Drive PX," in *Consumer Electronics Show*, January 2015. http://www.nvidia.com/object/drive-px.html.

[91] A. Teichman and S. Thrun, "Practical object recognition in autonomous driving and beyond," in *Advanced Robotics and its Social Impacts (ARSO), 2011 IEEE Workshop on*, pp. 35–38, Oct 2011.

[92] A. Teichman, J. Levinson, and S. Thrun, "Towards 3D object recognition via classification of arbitrary object tracks," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 4034–4041, May 2011.

[93] E. Ilg, R. Kuummerle, W. Burgard, and T. Brox, "Reconstruction of rigid body models from motion distorted laser range data using optical flow," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pp. 4627–4632, May 2014.

[94] D. Held, J. Levinson, S. Thrun, and S. Savarese, "Combining 3D Shape, Color, and Motion for Robust Anytime Tracking," in *Proceedings of Robotics: Science and Systems*, (Berkeley, USA), July 2014.

[95] I. Miller, M. Campbell, and D. Huttenlocher, "Efficient Unbiased Tracking of Multiple Dynamic Obstacles Under Large Viewpoint Changes," *Robotics, IEEE Transactions on*, vol. 27, pp. 29–46, Feb 2011.

[96] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. The MIT Press, 2006.

[97] C. Urmson, "The Google Self-Driving Car Project," tech. rep., Google Inc., 2011.

[98] J. Dickmann, N. Appenrodt, and C. Brenk, "How We Gave Sight to the Mercedes Robotic Car," in *IEEE Spectrum, Transportation Website*, Jul 2014. http://spectrum.ieee.org/transportation/self-driving/how-we-gave-sight-to-the-mercedes-robotic-car.

[99] T. Vanderbilt, "Let the Robot Drive The Autonomous Car of the Future is Here," in *Wired Magazine*, Jan 2012. http://www.wired.com/2012/01/ff_autonomouscars/.

[100] B. Ristic, N. Gordon, and A. Bessell, "On target classification using kinematic data," *Information Fusion*, vol. 5, pp. 15–21, 2004.

[101] G. Monteiro, C. Premebida, P. Peixoto, and U. Nunes, "Tracking and classification of dynamic obstacles using laser range finder and vision," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2006.

[102] Y. Freund and R. E. Schapire, "A Short Introduction to Boosting," *Journal of Japanese Society for Artificial Intelligence*, vol. 14, pp. 771–780, Sept 1999.

[103] K. V. Ramachandra, *Kalman Filtering Techniques for Radar Tracking*. Marcel Dekker, Inc., 2000.

[104] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., 2001.

[105] B. Vo and W. Ma, "The Gaussian Mixture Probability Hypothesis Density Filter," *IEEE: Transactions Signal Processing*, vol. 54, no. 11, pp. 4091–4104, 2006.

[106] K. Matzen and N. Snavely, "NYC3DCars: A Dataset of 3D Vehicles in Geographic Context," in *Proc. Int. Conf. on Computer Vision*, 2013.

[107] J. R. Schoenberg, M. Campbell, and I. Miller, "Posterior representation with a multi-modal likelihood using the gaussian sum filter for localization in a known map," *Journal of Field Robotics*, vol. 29, pp. 240–257, Mar 2012.

[108] I. Miller, M. Campbell, and D. Huttenlocher, "Map-aided localization in sparse global positioning system environments using vision and particle filtering," *Journal of Field Robotics*, vol. 28, pp. 619–643, Sept 2011.

[109] I. Miller, M. Campbell, D. Huttenlocher, F.-R. Kline, A. Nathan, S. Lupashin, J. Catlin, B. Schimpf, P. Moran, N. Zych, E. Garcia, M. Kurdziel, and H. Fujishima, "Team Cornell's Skynet: Robust perception and planning in an urban environment," *Journal of Field Robotics*, vol. 25, no. 8, pp. 493–527, 2008.

[110] I. Miller and M. Campbell, "Sensitivity Analysis of a Tightly-Coupled GPS/INS System for Autonomous Navigation," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 48, pp. 1115–1135, Apr 2012.

[111] A. Angelova, A. Krizhevsky, and V. Vanhoucke, "Pedestrian detection with a large-field-of-view deep network," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pp. 704–711, May 2015.

[112] Z. Cai, M. J. Saberian, and N. Vasconcelos, "Learning complexity-aware cascades for deep pedestrian detection," *CoRR*, vol. abs/1507.05348, 2015.

[113] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *International Journal of Computer Vision (IJCV)*, vol. 88, June 2010.

[114] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets Robotics: The KITTI Dataset," *International Journal of Robotics Research (IJRR)*, 2013.

[115] R. B. Girshick, P. F. Felzenszwalb, and D. McAllester, "Discriminatively Trained Deformable Part Models, Release 5." http://people.cs.uchicago.edu/ rbg/latent-release5/, 2012.

[116] P. E. Ross, "Nvidia Wants to Build the Robocar's Brain," in *IEEE Spectrum, Cars that Think Website*, Apr 2015. http://spectrum.ieee.org/cars-that-think/transportation/self-driving/nvidia-wants-to-build-the-robocars-brain.

[117] D. Gatziolis and H.-E. Andersen, "A Guide to LIDAR Data Acquisition and Processing for the Forests of the Pacific Northwest," Tech. Rep. PNW-GTR-768, United States Department of Agriculture: Forest Service Pacific Northwest Research Station, July 2008.

[118] R. Tamburo, E. Nurvitadhi, A. Chugh, M. Chen, A. Rowe, T. Kanade, and S. Narasimhan, "Programmable automotive headlights," in *European Conference of Computer Vision (ECCV)*, vol. 8692 of *Lecture Notes in Computer Science*, pp. 750–765, Springer International Publishing, 2014.